



UNIVERSIDAD  
DE MÁLAGA



E.T.S. INGENIERÍA  
**INFORMÁTICA**  
UNIVERSIDAD DE MÁLAGA

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA  
INFORMÁTICA GRADUADO EN INGENIERÍA INFORMÁTICA

---

## DIARIO DE DOLOR CATCH MY PAIN

---

Realizado por  
Javier Olea Aguilar

Tutorizado por  
José Luis Pastrana Brincones

Departamento  
Lenguajes y Ciencias de la Computación

UNIVERSIDAD DE MÁLAGA

MÁLAGA, 23 de septiembre de 2021

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA  
GRADO EN INGENIERÍA INFORMÁTICA

**Diario de Dolor**

**Catch my Pain**

Realizado por

**Javier Olea Aguilar**

Tutorizado por

**José Luís Pastrana Brincones**

Departamento

**Lenguajes y Ciencias de la Computación**

UNIVERSIDAD DE MÁLAGA  
MÁLAGA, SEPTIEMBRE DE 2021

Fecha defensa: 5 de octubre de 2021

# Resumen

En esta memoria de trabajo de fin de grado se presenta el proceso de desarrollo y de creación de la aplicación móvil “Diario de Dolor”.

Con esta aplicación se busca facilitar el proceso de consultas para los pacientes que padecen dolores crónicos, unificando todos los dolores que ha padecido en el pasado y que padece en la actualidad el usuario en un único informe generado por la aplicación. Mencionado informe podrá ser entregado directamente por el paciente a su doctor o médico de cabecera lo que le permitirá observar con todo detalle la evolución de los dolores padecidos por el paciente a lo largo del tiempo, además de visualizar información médica útil para la posible resolución del tratamiento.

Los usuarios registrados en la aplicación tendrán que introducir sus datos personales tales como la fecha de nacimiento, peso, altura y nombre completo; además de completar un cuestionario de salud básico. Todo esto busca aportar información médica relevante que podría ser de gran utilidad en una futura búsqueda de tratamiento por parte del especialista clínico a su cargo.

Una vez se hayan registrado todos estos datos en la aplicación, el usuario podrá ingresar su historial de dolores padecidos de una forma rápida e intuitiva. Ya con todos los datos ingresados se podrán generar informes completos y con todo tipo de detalles.

La aplicación ha sido implementada con el lenguaje de programación Java, mediante la herramienta de trabajo de Android Studio. Los datos se almacenarán en una base de datos relacional con SQLite.

**Palabras clave:** *informe, dolores crónicos, Android Studio, Java, SQL.*



# Abstract

This end-of-degree work report presents the process of development and creation of the mobile application “Diario de Dolor”.

This application seeks to facilitate the consultation process for patients suffering from chronic pain, unifying all the pain that the user has suffered in the past and that currently suffers in a single report generated by the application. The aforementioned report may be delivered directly by the patient to his doctor or nurse, which will allow him to observe in detail the evolution of the pain suffered by the patient over time, in addition to displaying useful medical information for the possible resolution of the treatment.

Registered users in the application will have to enter their personal data such as date of birth, weight, height and full name; in addition to completing a basic health questionnaire. All this seeks to provide relevant medical information that could be very useful in a future search for treatment by the clinical specialist in charge.

Once all these data have been registered in the application, the user will be able to enter their history of pain suffered in a fast and intuitive way. With all the data entered, it will be possible to generate complete reports with all kinds of details.

The application has been implemented with the Java programming language, using the Android Studio work tool. The data will be stored in a relational database with SQLite.

**Keywords:** *report, chronic pain, Android Studio, Java, SQL.*



# Índice general

<b>1. Introducción</b>	<b>13</b>
1.1. Estructura de la memoria . . . . .	13
1.2. Motivación . . . . .	14
1.3. Objetivos . . . . .	15
1.4. Estado del arte . . . . .	16
1.4.1. Fibroline . . . . .	16
1.4.2. My Pain Diary & Symptom Tracker . . . . .	17
1.4.3. Painometer . . . . .	18
1.4.4. Pain Care App . . . . .	19
1.5. Tecnologías a utilizar . . . . .	20
1.5.1. Látex y Texmaker . . . . .	20
1.5.2. UML . . . . .	21
1.5.3. Visual Paradigm . . . . .	21
1.5.4. Android Studio . . . . .	22
1.5.5. Java . . . . .	22
1.5.6. XML . . . . .	23

1.5.7. SQLite . . . . .	23
1.6. Metodología empleada . . . . .	24
<b>2. Análisis</b>	<b>27</b>
2.1. Actores y participantes . . . . .	27
2.2. Obtención de requisitos . . . . .	28
2.2.1. Requisitos funcionales . . . . .	29
2.2.2. Requisitos no funcionales . . . . .	32
2.3. Casos de uso . . . . .	32
<b>3. Diseño</b>	<b>47</b>
3.1. Arquitectura . . . . .	47
3.2. Diseño de la base de datos . . . . .	48
3.2.1. Diagrama Entidad-Relación . . . . .	48
3.2.2. Entidades y Relaciones de la base de datos . . . . .	49
3.2.3. Diagrama de clases . . . . .	49
3.2.4. Operaciones y Relaciones estáticas de la base de datos . . . . .	50
3.3. Implementación . . . . .	51
3.3.1. Implementación primera iteración . . . . .	52
3.3.2. Implementación segunda iteración . . . . .	54
3.3.3. Implementación tercera iteración . . . . .	55
3.3.4. Implementación cuarta iteración . . . . .	56
<b>4. Conclusión</b>	<b>59</b>



4.1. Conclusiones . . . . .	59
4.2. Líneas futuras . . . . .	60



# Índice de cuadros

2.1. Requisitos funcionales . . . . .	29
2.2. Requisitos no funcionales . . . . .	32
2.3. Caso de uso RF-01 . . . . .	34
2.4. Caso de uso RF-02 . . . . .	35
2.5. Caso de uso RF-03 . . . . .	36
2.6. Caso de uso RF-04 . . . . .	37
2.7. Caso de uso RF-05 . . . . .	38
2.8. Caso de uso RF-06 . . . . .	38
2.9. Caso de uso RF-07 . . . . .	39
2.10. Caso de uso RF-08 . . . . .	39
2.11. Caso de uso RF-09 . . . . .	40
2.12. Caso de uso RF-10 . . . . .	41
2.13. Caso de uso RF-11 . . . . .	42
2.14. Caso de uso RF-12 . . . . .	43
2.15. Caso de uso RF-13 . . . . .	44
2.16. Caso de uso RF-14 . . . . .	45



# Índice de figuras

1.1. Fibroline: Capturas aplicación móvil . . . . .	17
1.2. My Pain Diary: Capturas aplicación móvil . . . . .	18
1.3. Painometer: Capturas aplicación móvil . . . . .	19
1.4. Pain Care App: Capturas aplicación móvil . . . . .	20
1.5. Latex: Logotipo . . . . .	20
1.6. Visual Paradigm: Logotipo . . . . .	21
1.7. Android Studio: Logotipo . . . . .	22
1.8. SQLite: Logotipo . . . . .	23
2.1. Diagrama casos de usos: Acceso a la aplicación . . . . .	33
2.2. Diagrama casos de usos: Usuario ya validado . . . . .	34
3.1. Diagrama Entidad-Relación - Diario de Dolor . . . . .	48
3.2. Diagrama Clases - Diario de Dolor . . . . .	50
3.3. Estructura proyecto en Android Studio . . . . .	52
3.4. Diario de Dolor: Pantalla principal . . . . .	53
3.5. Diario de Dolor: edición y creación de dolores . . . . .	55
3.6. Diario de Dolor: Pantalla informes . . . . .	57

1.	Diario de Dolor: Captura Pantalla Inicio de Sesión . . . . .	65
2.	Diario de Dolor: Capturas Registro de Usuarios . . . . .	66
3.	Diario de Dolor: Captura Panel principal . . . . .	67
4.	Diario de Dolor: Captura Perfil . . . . .	68
5.	Diario de Dolor: Capturas Dolores . . . . .	69
6.	Diario de Dolor: Capturas Informe . . . . .	70

# Capítulo 1

## Introducción

### 1.1. Estructura de la memoria

La memoria de trabajo se ha estructurado de modo que se siga un orden y una correlación tanto cronológica como estructural con las etapas del desarrollo del software, en este caso con un desarrollo iterativo [1]. De este modo se ha establecido la siguiente división de capítulos en el documento:

1. **Introducción:** En esta primera parte del informe se hace un breve acercamiento al proceso que se ha seguido para el desarrollo de la aplicación. Empezando con las metas que se persigue en este proyecto exponiendo su motivación y objetivos, así como la propia metodología y tecnologías utilizadas y que han sido vitales para la correcta finalización del proyecto.
2. **Análisis:** Ya comenzando con el estudio en profundidad del proyecto, se pasa a analizar los posibles casos de uso de la aplicación, así como su respectiva definición de los requisitos del sistema. También se tiene en cuenta y se entra en detalle con los actores que participarán en la aplicación y el alcance que tiene la misma.
3. **Diseño:** En este punto se especifica la arquitectura que tendrá el sistema, así como el diseño de la base de datos con sus respectivos diagramas.

4. **Implementación:** Se realiza una completa explicación del proceso del desarrollo de la aplicación exponiendo el diseño de la interfaz, el proceso de creación de la base de datos y la implementación de los puntos críticos tanto de su *backend* como de su *frontend*.
5. **Conclusión y líneas futuras:** Ya para terminar, se expondrán las conclusiones obtenidas en el proyecto, intentando proporcionar una visión más personal sobre mismo. Además se dejarán abiertas posibles líneas de trabajo futuras, que podrían ampliar y continuar el presente trabajo.

## 1.2. Motivación

Siempre he creído que la función principal de estos trabajos debe ser la de aprender y afianzar los conocimientos ya adquiridos en el grado, siendo el fin lucrativo algo muy secundario o incluso inexistente en la gran mayoría de los trabajos, como ocurre en mi caso. Por este motivo creo que el ámbito de la medicina es el más adecuado para la realización de los trabajos de fin de grado, puesto que el producto obtenido por ello podría ser de gran utilidad en el futuro para personas que realmente lo necesiten.

De este modo, en mi caso personal, he vivido de primera mano la experiencia de familiares cercanos que han padecido dolores crónicos durante mucho tiempo, lo que me ha hecho consciente de los reales y lo muy presentes que están en nuestro día a día, y de lo importante que es el consultarlos con especialistas del sector para buscarles un tratamiento adecuado.

En este punto es en el que entra esta herramienta, que busca facilitar la comunicación entre el paciente y los diferentes especialistas del sector gracias a los informes.

Mirando al pasado creo que realmente le hubiera sido de gran utilidad a mi familiar una aplicación como esta, ya que simplifica enormemente la comunicación con los especialistas, evitando el tener que repetir una y otra vez los mismos síntomas padecidos en cada visita a un especialista distinto, evitando además el posible olvido de alguno de ellos.

Con todo esto, creo que el trabajo ha sido una propuesta muy interesante que me ha servido para poner a prueba los conocimientos aprendidos durante todos estos años de carrera y que se podría contemplar como el culmen de todo las competencias adquiridas y el esfuerzo invertido



en estos años de estudios.

### 1.3. Objetivos

Con esta propuesta se persigue el facilitar el proceso de presentación de los síntomas de los dolores crónicos padecidos por el usuario de la aplicación hacia los posibles especialistas del sector clínico, ya sean terapeutas, doctores o médicos de cabecera.

La aplicación está destinada para las personas padecientes de los dolores crónicos, de modo que ellos rellenarán sus datos personales y el cuestionario clínico suministrado por la propia aplicación, además de registrar la totalidad de los dolores padecidos con todo tipo de detalle. Con toda esta información el usuario podrá generar un informe a través de la aplicación que recogerá todos los datos y que podrá ser entregado al especialista en cuestión.

Con respecto al formato de registro de los dolores, el usuario podrá registrar qué, dónde, cuándo y cuánto le duele para poder compartir esa información más fácilmente con el médico y no tener que repetir explicaciones en cada visita. El usuario podrá señalar con exactitud el lugar del dolor en una silueta del cuerpo humano presente en el registro de nuevos dolores en la aplicación, y especificar tanto la intensidad de dicho dolor como la especificación de los días, semanas o meses padecidos del dolor, así como las franjas horarias en las que están presentes.

Con la generación del informe se persigue lo siguiente:

- Asegurar la correcta comunicación entre paciente y especialista.
- Recoger todos los datos registrados en la aplicación por el usuario, con lo que se evite el posible olvido de algún síntoma o algún dato relevante, ayudando en la correcta prescripción del tratamiento por parte del especialista.
- Mostrar toda la información médica y los dolores del usuario registrados de una forma gráfica y fácil de leer para los especialistas médicos.

Como se puede intuir con lo dicho hasta ahora, el acceso por parte de los médicos no está planteado para esta aplicación, de modo que el único que podrá tener acceso de forma directa

a sus datos en la aplicación será el propio usuario registrado y logueado en ella. De este modo el usuario será el encargado de generar su propio informe, siempre de una forma instantánea y automática por parte de la aplicación, y entregarle dicho informe al especialista en cuestión, ya sea en el formato generado por la aplicación o con una impresión física a papel. Este último punto dependerá de las preferencias por parte del usuario.

## **1.4. Estado del arte**

Ahora pasaremos a hablar del estado actual del sector en el nicho concreto de la medicina concerniente a los dolores crónicos, de modo que se expondrán las aplicaciones y páginas web ya creadas e implementadas en el mercado.

De este modo se expondrán las funcionalidades de estas aplicaciones, resaltando tanto las similitudes como los puntos que diferenciarán a estas aplicaciones de nuestra aplicación "Diario de Dolor".

### **1.4.1. Fibroline**

Aplicación móvil dirigida a paliar la intensidad los dolores crónicos, intentando buscar tratamientos o hábitos saludables para mejorar la calidad de vida de sus usuarios y mantener a raya los dolores sufridos [2].

De este modo la aplicación no busca mantener un historial de los dolores, si no que únicamente pretende buscar un remedio para paliar dichos dolores. Por esto mismo la aplicación no genera ningún tipo de informe ni nada parecido, ya que en ningún momento busca interactuar con los médicos.



(a) Captura 1



(b) Captura 2

Figura 1.1: Fibroline: Capturas aplicación móvil

### 1.4.2. My Pain Diary & Symptom Tracker

Aplicación móvil de pago que maneja el historial de dolores de sus usuarios, de modo que les permite registrar la localización de sus dolores crónicos así como su intensidad y duración. Adicionalmente también busca facilitar la comunicación entre pacientes y doctores mediante la generación de informes.

Pese a ser de pago, se puede considerar como una de las aplicaciones de salud más exitosas del mercado, puesto que, a día 11-03-2021, se encuentra en la posición #83 en el ranking de aplicaciones médicas de la App Store.

Como podemos ver, esta aplicación tiene unos objetivos muy similares a la nuestra, por lo que siendo una aplicación ya consagrada y exitosa en su ámbito, la tendremos como el espejo donde mirarnos, extrayendo posibles ideas de la misma.

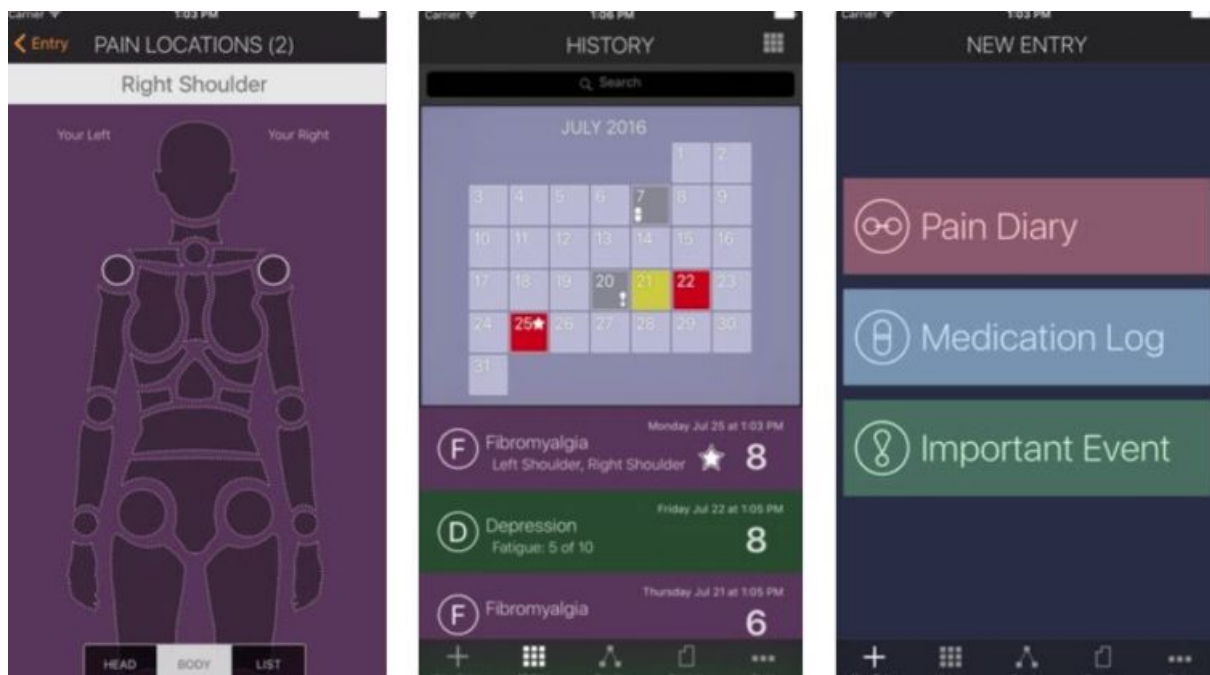


Figura 1.2: My Pain Diary: Capturas aplicación móvil

### 1.4.3. Painometer

Aplicación móvil que sirve para medir y registrar la intensidad del dolor a lo largo del tiempo, ganadora además del premio mHealth - Mobile World Capital Barcelona.

Se podría decir que es una versión españolizada de "My Pain Diary", con la diferencia que es gratuita pero a la vez incluye muchas menos funcionalidades que ésta. Por ejemplo la generación de informes se realiza de una forma más simple, con una información algo escueta y todo mediante el correo electrónico.

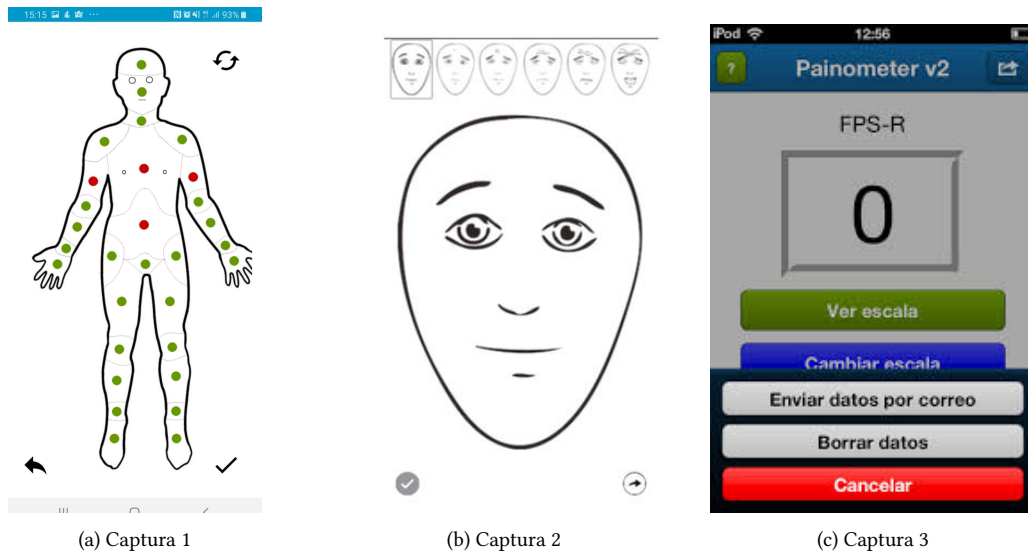


Figura 1.3: Painometer: Capturas aplicación móvil

#### 1.4.4. Pain Care App

Aplicación gratuita de la App Store que busca paliar la intensidad de los dolores crónicos de sus usuarios, de modo que los propios usuarios buscan a través de la aplicación un tipo de ejercicio de relajación o de control de respiración que les ayude para amenizar los dolores.

En este caso, aunque ambas aplicaciones pertenezcan al mismo ámbito de la medicina, vemos como las diferencias entre esta y nuestra aplicación son grandes. No se mantiene un historial de los dolores ni busca hacer de intermediario con los doctores, ya que ella misma trata de buscar tratamientos para los dolores.

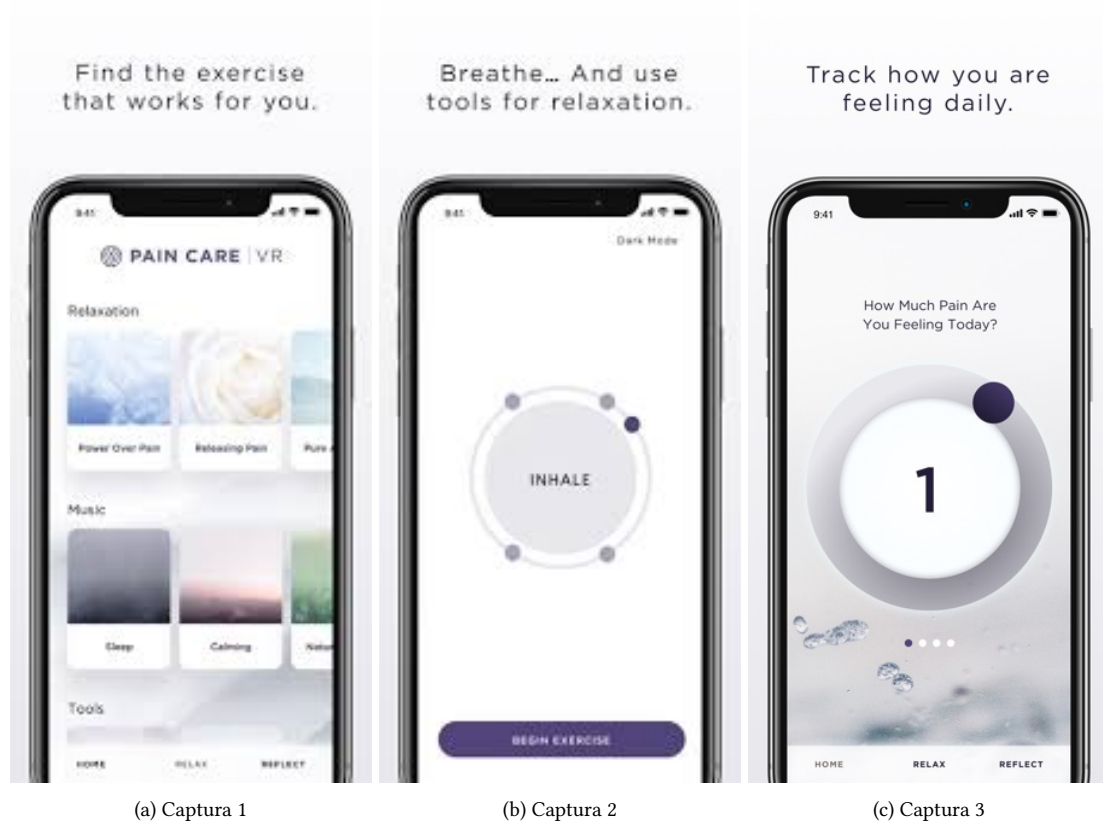


Figura 1.4: Pain Care App: Capturas aplicación móvil

## 1.5. Tecnologías a utilizar

Ahora se describirá las distintas tecnologías, herramientas y lenguajes que se han empleado para la realización al completo del proyecto.

### 1.5.1. Látex y Texmaker

Para la creación de este presente documento se ha empleado el lenguaje de bajo nivel Látex [3] [4], buscando un procesador de texto profesional a la par que elegante y agradable para el lector. Siempre siguiendo las plantillas como la estructura y las normas de edición especificadas por la Universidad de Málaga.



Figura 1.5: Latex: Logotipo

Como editor de texto se ha utilizado Texmaker, ya que aporta muchas herramientas útiles para la creación de documentos con Látex, siendo a la vez una herramienta intuitiva y con funcionalidades muy útiles como la pre-visualización del documento compilado en PDF.

### 1.5.2. UML

Como lenguaje de modelado software emplearemos UML (Lenguaje Unificado de Modelado), ya que es el lenguaje más conocido y utilizado en la actualidad en todo lo relacionado con la visualización y especificación de proyectos software [5].

Como veremos posteriormente este modelo será utilizado para la realización tanto de diagramas de casos de uso como esquemas de bases de datos, entre otros.

### 1.5.3. Visual Paradigm

Herramienta online que permite la creación de tablas y de diagramas, tanto UML como diagramas de flujo o de mapas de procesos. Destaca, entre otras muchas cosas, por su espacio de trabajo situado al completo en la nube, permitiendo el uso de la herramienta sin ningún tipo de descarga.



En este proyecto se ha empleado para diversos diagramas, entre ellos, para la creación de diagramas de casos de usos, permitiendo una visualización gráfica de los actores que interactuarán con el sistema y de las acciones que podrán realizar los mismos dentro de la aplicación.

Figura 1.6: Visual Paradigm: Logotipo

Por otro lado también se ha hecho uso de esta herramienta para la creación tanto del modelo entidad relación de la base de datos como para el diagrama de clases del mismo. Ambos diagramas se verán más adelante en el apartado de diseño 3.2.

### 1.5.4. Android Studio



Figura 1.7: Android Studio:  
Logotipo

Entorno de desarrollo integrado para aplicaciones móviles con sistema operativo Android. Consta de una sencilla e intuitiva interfaz que la convierten en una herramienta fácil de utilizar [6].

Como grandes ventajas de esta herramienta tenemos la posibilidad de programar en distintos lenguajes (C++, Java, Go, etc.) y su potente emulador que permite comprobar el estado actual de la aplicación en el momento.

Además es la IDE oficial de Android, lo que nos asegura el correcto funcionamiento del software, ya que es la misma herramienta que usan los propios desarrolladores de Android.

### 1.5.5. Java

De entre todos los lenguajes que soporta Android Studio, se ha elegido el lenguaje orientado a objetos Java [7].

Uno de los motivos de esta elección es la gran experiencia que he obtenido durante estos años de carrera en este lenguaje, ya que ha sido, sin lugar a dudas, el lenguaje más usado en el grado.

Pero este no ha sido el único motivo, ya que Java nos ofrece todas funcionalidades que nos resultan imprescindibles para la creación de este software. Como ejemplo está el hecho de que es un lenguaje orientado a objetos, algo vital, en mi opinión, para la creación de aplicaciones móviles, además de ser un lenguaje sencillo e independiente de la plataforma, lo que permite ejecutarse en cualquier dispositivo sin ningún tipo de problema.

Por otro lado, este lenguaje consta con innumerables paquetes y librerías que amplían aún más las capacidades y posibilidades que nos ofrece el lenguaje, convirtiéndolo en uno de los lenguajes más completos del momento.



### 1.5.6. XML

Así como se ha utilizado Java como lenguaje de programación para el *backend* de la aplicación, para el *frontend* de la misma se ha hecho uso de XML [8].

Este lenguaje formado por etiquetas ofrece un gran abanico de posibilidades permitiendo tanto etiquetas individuales como anidadas permitiendo una gran diversidad a la hora de diseñar la interfaz de los sistemas.

Además mencionar que Android Studio tiene totalmente integrado este lenguaje, de modo que incluso ofrece la funcionalidad de, para ciertas modificaciones, hacer cambios visuales de la estructura y formato de la página desde la misma pestaña del diseño de las pantallas.

### 1.5.7. SQLite

Motor de base de datos SQL autónomo y de configuración simple sin servidor, reduciendo de forma significativa los esfuerzos de administración de las bases de datos con respecto a otros motores como mySQL.

SQLite [9] almacena toda la base de datos en un archivo único multiplataforma, siendo este punto una gran ventaja en cuanto a temas de seguridad y migración, lo que lo convierte en el gestor de base de datos ideal para aplicaciones móviles en Android.

Por todo esto SQLite se ha convertido en el gestor de base de datos principal de Android Studio y, por ende, en el motor de datos que usaremos para este proyecto.



Figura 1.8: SQLite: Logotipo

## 1.6. Metodología empleada

Para el análisis y diseño del sistema de información se utiliza el lenguaje de especificación UML. La programación de la aplicación se realiza mediante el uso de programación orientada a objetos, usando como herramienta de programación Android Studio y el lenguaje de Programación Java.

El desarrollo del proyecto se ha llevado a cabo mediante una metodología iterativa incremental [1]. En la que, en cada iteración, se han añadido más funcionalidades, para ir creando cada vez versiones más complejas y completas del sistema. Con esto también se ha buscado obtener retroalimentación del usuario lo antes posible, para orientar el desarrollo al cumplimiento de las necesidades del usuario y realizar todas las posibles adaptaciones para cumplir con todos los objetivos planteados.

Otra razón para el uso de una metodología iterativa e incremental, ha sido la posibilidad de obtener productos funcionales con parte de las características solicitadas, en fases tempranas y a la capacidad de adaptabilidad, lo que ha favorecido a satisfacción al completo todas las necesidades del proyecto.

Más concretamente, se han propuesto 4 iteraciones de 2 semanas cada una del siguiente modo:

Iteración 1: Puesta a punto de la base de datos y primera configuración de la aplicación. Se implementa el registro de usuarios con su respectivo formulario de salud que deberán rellenar en el mismo proceso de registro. Además, se seleccionan los colores corporativos y el diseño de la barra de herramientas y de los botones presentes en la aplicación.

Iteración 2: Se implementan las pantallas y las funcionalidades para la edición de los perfiles, de modo que una vez registrado e iniciado sesión podrá acceder a sus propios datos personales y a su formulario de salud, pudiendo modificarlos en cualquier momento.

Iteración 3: Con toda la parte del usuario implementada, se pasa a la creación de las pantallas y las funcionalidades correspondientes para la creación y la edición de los dolores

crónicos.

Iteración 4: Ya como último paso se implementa la creación de los informes mediante la recogida de todos los datos almacenados en la aplicación. Además, en esta etapa se realizan las últimas pinceladas y correcciones de la aplicación, terminando en esta iteración con la aplicación final totalmente funcional.



# Capítulo 2

## Análisis

### 2.1. Actores y participantes

En este punto plasmaremos los tipos de roles que interactuarán de forma directa o indirecta con la aplicación, especificando los diferentes privilegios o similitudes que pudiera haber entre los mismos, si es que las hubiera.

Como ya se ha ido viendo hasta ahora, el método de acceso a la aplicación es muy simple, de modo que todo usuario registrado y que tiene acceso a la aplicación, ejerce el rol de paciente o de convaleciente de los dolores crónicos. De este modo hablaremos de estos usuarios como los actores principales de la aplicación, ya que son los que tendrán acceso directo a la misma, y hablaremos de los especialistas médicos como los actores secundarios, ya que no dispondrán de este acceso directo a la aplicación.

Los actores principales o pacientes, serán los encargados de registrarse en la aplicación, introduciendo sus datos personales y rellenando el formulario de salud de la aplicación, pudiendo registrar todo tipo de dolores crónicos que padezcan. De este modo tendrán acceso en todo momento a sus datos, pudiendo modificarlos o añadiendo nuevos tipos de dolores en cualquier momento y pudiendo generar, mediante la aplicación, un informe completo y conciso con todo tipo de datos clínicos sobre el paciente que pueden ser de gran utilidad para la búsqueda de un tratamiento.

En este punto es en el que entra en escena los actores secundarios o los especialistas mé-

dicos, a los que sus pacientes les llevarán estos informes en busca de un tratamiento para sus dolores crónicos. Los informes les podrán ser suministrados en el formato digital (PDF) generado por la propia aplicación o con el informe impreso de forma física, dependiendo de las preferencias de los usuarios. Así serán los doctores, con ayuda de los informes, los que buscarán un tratamiento efectivo para los dolores padecidos por los usuarios.

De este modo los médicos no tendrán acceso directo de ningún tipo a la aplicación, pero si que tendrán un papel imprescindible en la misma, ya que, a fin de cuentas, lo que la aplicación busca es la correcta comunicación entre pacientes y médicos, intentando facilitar lo máximo posible este proceso para los pacientes.

## 2.2. Obtención de requisitos

Aquí da comienzo una de las etapas más importantes en el proceso de desarrollo del software, ya que un correcto análisis y refinamiento de los requisitos del sistema nos facilitará enormemente la correcta implementación del mismo.

Se busca realizar una especificación completa de los requisitos y de este modo contemplar las funcionalidades, datos, rendimiento, interfaz y restricciones que abarcarán el software. Además especificaremos entre dos tipos de requisitos, los funcionales y los no funcionales [10].

Los **requisitos funcionales** nos describirán las interacciones entre el sistema y su ambiente, incluyendo tanto usuarios como cualquier otro tipo de sistema externo que interactúe con el mismo. De este modo, estos requisitos nos especificarán lo que el sistema debe hacer una vez este completo y funcional.

Los **requisitos no funcionales** nos definirán los atributos del sistema o incluso los del ambiente del sistema. Estos requisitos también son conocidos como los atributos de calidad del sistema, ya que nos dicen características de funcionamiento del software.

Sin embargo, como ya se ha mencionado anteriormente en el apartado 1.6, la metodología empleada será la iterativa incremental, por lo que, aunque en este punto veamos todos los requisitos al completo, estos se irán implementando en su respectiva iteración y no todos de

golpe. Para esto se especificará posteriormente que requisitos abarcarán cada iteración.

### 2.2.1. Requisitos funcionales

Cuadro 2.1: Requisitos funcionales

<b>Id</b>	<b>Título</b>	<b>Descripción</b>	<b>Observaciones</b>
RF-01	Iniciar sesión	Se pedirá tanto el correo del usuario como la contraseña para el inicio de sesión.	Los identificadores de usuario y los correos son únicos.
RF-02	Validar acceso	Comprobar quién accede a la plataforma mediante identificador y contraseña.	Se validará que tanto el correo como la contraseña son correctos y, en ese caso, se permitirá el acceso a la aplicación.
RF-03	Registro	Un sistema de registro en la plataforma para usuarios nuevos.	Se pedirán los datos del usuario como podrán ser su nombre, apellidos, fecha de nacimiento, peso, altura, teléfono y los correspondientes al cuestionario de salud
RF-04	Realización cuestionario básico de salud	Al crear un nuevo usuario, se le mostrará el cuestionario de salud creado por la Consejería de Salud de la Junta de Andalucía.	Para dicho cuestionario se podrán suprimir algunas preguntas que no tendrían ningún tipo de relación con la patología de los dolores, buscando amenizar el proceso de creación de un nuevo usuario.
RF-05	Cerrar sesión	Terminación de la sesión del usuario.	
RF-06	Modificar perfil	Se permitirá modificar información correspondiente al usuario.	La modificación se podrá realizar tanto para los datos personales, como para los datos correspondientes al cuestionario de salud.

*Continúa en siguiente página*

Cuadro 2.1 – Continuada desde página anterior

<b>Id</b>	<b>Título</b>	<b>Descripción</b>	<b>Observaciones</b>
RF-07	Historial de dolores	Una vez iniciado sesión, el usuario podrá ver un listado con todos los dolores que ha añadido a la aplicación.	Esta información será totalmente privada y únicamente accesible por el usuario en cuestión.
RF-08	Añadir un nuevo dolor	El usuario podrá añadir un nuevo registro de dolor a su historial de dolores en cualquier momento.	Para su registro será necesario especificar el dónde, cuándo y cuánto del dolor
RF-09	Modificar dolor	Los registros de dolores ya añadidos a la aplicación podrán ser modificados	Estas modificaciones pueden ser causadas por un fallo por parte del usuario al registrar un dolor o por alguna variación en el mismo.
RF-10	Eliminar dolor	La eliminación al completo de un registro de dolor en la aplicación.	Esta aplicación busca seguir el registro histórico de dolores de un paciente, por lo que esta función no será la solución de dolores ya pasados. Está destinado a la eliminación de dolores añadidos por error o con valores erróneos.
RF-11	Intensidad del dolor	Se podrá especificar la intensidad de los dolores mediante el color, yendo desde tonalidades amarillas para las más leves a tonalidades rojas para las más intensas.	Esto se registrará en la aplicación mediante una escala de dolor, en la que el usuario deberá seleccionar en que escala de intensidad se encuentra el dolor.
RF-12	Estado de los dolores	Este atributo nos dirá cuándo ha tenido lugar el dolor.	El estado nos representará si el dolor lo está padeciendo actualmente de forma activa el usuario, si el dolor lo padece de forma intermitente o si es un dolor ya pasado y no lo padece actualmente.

*Continúa en siguiente página*



Cuadro 2.1 – Continuada desde página anterior

<b>Id</b>	<b>Título</b>	<b>Descripción</b>	<b>Observaciones</b>
RF-13	Lugar del dolor	Para esto haremos uso de siluetas del cuerpo humano, dónde el usuario podrá localizar con exactitud su dolor.	Serán necesarias un total de dos siluetas del cuerpo humano para cada género, una para la parte frontal del cuerpo y otra para la trasera. Haciendo un total de cuatro siluetas 2D del cuerpo humano. Sobre estas plantillas se guardarán las coordenadas del lugar del dolor.
RF-14	Generar informe de dolores	El usuario podrá generar un informe que se creará de forma automática con todos los datos del usuario y los dolores que haya añadido a la aplicación.	El informe seguirá una plantilla y podrá servir como soporte para los posibles doctores o enfermeros visitados por el usuario.

### 2.2.2. Requisitos no funcionales

Cuadro 2.2: Requisitos no funcionales

<b>Id</b>	<b>Título</b>	<b>Descripción</b>
RNF-01	Seguridad	La información estará protegida ante accesos no autorizados mediante mecanismos de validación.
RNF-02	Encriptación de contraseñas	Todas las contraseñas deben ser privadas y solo conocidas por el usuario. Para esto se almacenarán todas las contraseñas de forma cifrada en la base de datos.
RNF-03	Confiabilidad	Un usuario solo podrá tener acceso a su información. En ningún caso podrá acceder a información correspondiente a otro usuario.
RNF-04	Diseño intuitivo	Debido al ámbito que abarca la aplicación, podrá tener usuarios de una edad muy variada incluyendo personas ancianas sin mucha experiencia con las nuevas tecnologías. Por este motivo el diseño de la aplicación debe ser claro, sencillo, directo e intuitivo.
RNF-05	Rendimiento	Tener un tiempo de respuesta bajo y una buena eficiencia en el manejo de los recursos.
RNF-06	Escalabilidad	Posibilidad de ampliación de la aplicación.

### 2.3. Casos de uso

Los casos de uso expresan una descripción no técnica de las acciones y los escenarios que pueden tener lugar en el sistema. Cada caso de uso refleja una actividad determinada que puede realizar un usuario con el objetivo de alcanzar una meta específica. A esto también lo podemos denominar como una característica del sistema.

Para la estructura de los casos de uso, se seguirá las especificaciones suministradas por el

Marco de Desarrollo de la Junta de Andalucía [11], de modo que cada caso de uso contará con su descripción, pre-condición, post-condición, escenario principal y escenario alternativo.

Como se ha mencionado en el apartado 2.1, solo habrá un único tipo de actor en la aplicación que tendrá acceso al sistema, por lo que los diagramas de casos de uso girarán al completo sobre este tipo de usuario. De este modo, dividiremos los diagramas de los casos de usos en dos escenarios, uno primero que tiene como meta el acceso a la aplicación, y un segundo diagrama en el que se plantea al usuario ya validado y con acceso al completo al sistema. Veremos ambos diagramas a continuación:

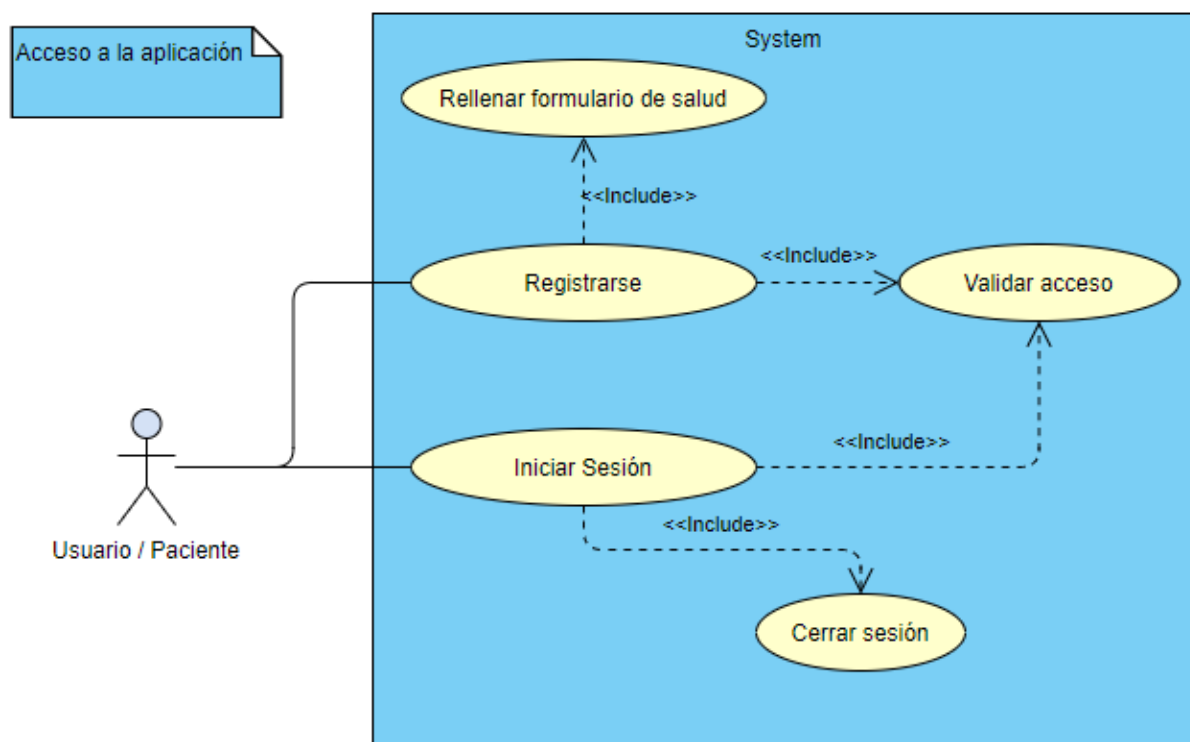


Figura 2.1: Diagrama casos de usos: Acceso a la aplicación

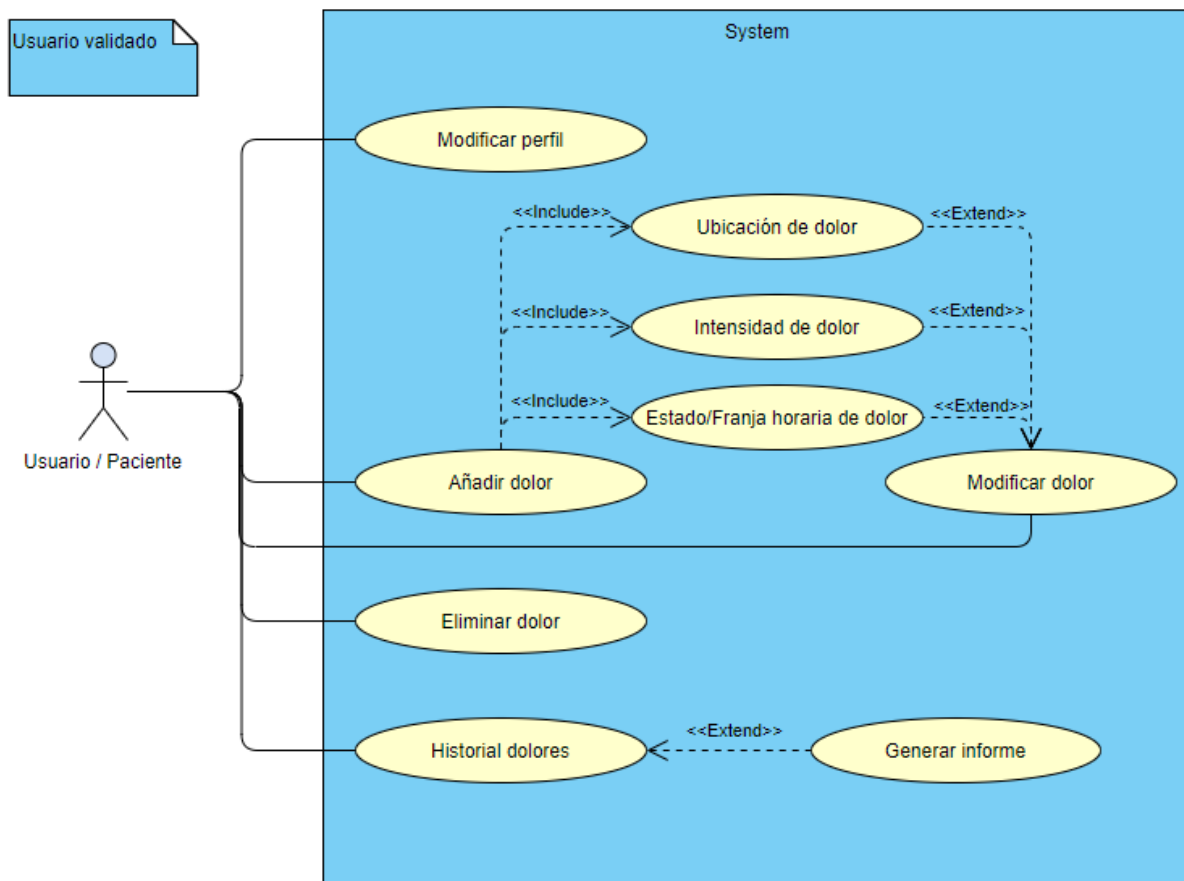


Figura 2.2: Diagrama casos de usos: Usuario ya validado

Ahora pasaremos a una descripción más detallada de los casos de uso, incluyendo ya los requisitos a los que satisfacen y haciendo una explicación de los mismos. Para ello, como se ha mencionado anteriormente, se seguirá la plantilla establecida por MADEJA [11]. Mencionar que al encontrarnos ante un sistema al que solo tiene acceso un tipo de usuario, se asumirá que todos los casos de usos se especifican sobre este mismo tipo de actor y no será necesario especificarlo en cada descripción.

Cuadro 2.3: Caso de uso RF-01

<b>Caso de uso</b>	RF-01 Iniciar Sesión en aplicación
<b>Descripción</b>	Un usuario registrado en Diario de Dolor, podrá iniciar sesión en la aplicación.
<b>Pre-condición</b>	El sistema funciona correctamente y el actor esta correctamente registrado y aún no ha iniciado sesión.

*Continúa en siguiente página*

<b>Post-condición</b>	El sistema permite el acceso y muestra al usuario la pantalla de inicio de su perfil.
<b>Escenario principal</b>	
<ol style="list-style-type: none"> <li>1. El actor accede al sistema y este muestra la pantalla de inicio de sesión.</li> <li>2. El usuario introduce su nombre de usuario y contraseña y le da al botón de “Iniciar sesión”.</li> <li>3. El sistema comprueba que los datos son correctos.</li> <li>4. Se permite el acceso al actor mostrándole su correspondiente pantalla de perfil.</li> </ol>	
<b>Escenario alternativo</b>	
<p>2.b No se han introducido todas las credenciales o las credenciales introducidas no se encuentran en el sistema.</p> <p>2.c El sistema muestra un mensaje de error por credenciales incorrectos.</p>	

Cuadro 2.4: Caso de uso RF-02

<b>Caso de uso</b>	RF-02 Validar acceso
<b>Descripción</b>	El sistema podrá comprobar si los credenciales introducidos están registrados en la aplicación.
<b>Pre-condición</b>	El sistema funciona correctamente y el actor introduce sus credenciales.
<b>Post-condición</b>	El sistema confirma si las credenciales son correctas o no.
<b>Escenario principal</b>	
<ol style="list-style-type: none"> <li>1. El actor introduce sus credenciales y pulsa el botón de “Iniciar Sesión” para acceder a la aplicación.</li> <li>2. El sistema comprueba que tanto el nombre de usuario como la contraseña son correctos.</li> <li>3. Se permite el acceso al sistema.</li> </ol>	

*Continúa en siguiente página*

<b>Escenario alternativo</b>
2.b Las credenciales introducidas no se encuentran en el sistema.
2.c El sistema muestra un mensaje de error por credenciales incorrectos.
3.a Por problemas de conexión el sistema no puede comprobar los credenciales introducidos.
3.b El sistema muestra un mensaje de error por pérdida de conexión.

Cuadro 2.5: Caso de uso RF-03

<b>Caso de uso</b>	RF-03 Registrarse en aplicación
<b>Descripción</b>	Un usuario que quiera acceder a la aplicación puede registrarse.
<b>Pre-condición</b>	El sistema funciona correctamente y el actor aún no está registrado en ella.
<b>Post-condición</b>	El actor queda registrado en la aplicación.
<b>Escenario principal</b>	
<ol style="list-style-type: none"> <li>1. El actor accede a la aplicación y pulsa el botón de “Registrarse”.</li> <li>2. El actor introduce todos los datos personales que le pide el sistema, así como el nombre de usuario y contraseña.</li> <li>3. El actor rellena el formulario de salud.</li> <li>4. El usuario pulsa el botón de “Aceptar” y confirma la creación del usuario.</li> </ol>	
<b>Escenario alternativo</b>	

*Continúa en siguiente página*

2.b El usuario ya existe en el sistema.
2.c El sistema muestra un mensaje de error por usuario ya registrado.
4.a El usuario ha dejado datos sin rellenar.
4.b El sistema muestra un mensaje de error por falta de datos.

Cuadro 2.6: Caso de uso RF-04

<b>Caso de uso</b>	RF-04 Realizar cuestionario básico de salud
<b>Descripción</b>	Al registrarse el actor, deberá rellenar un formulario básico de salud.
<b>Pre-condición</b>	El actor se está registrando en la aplicación y ya ha introducido sus datos personales, así como el nombre de usuario y contraseña.
<b>Post-condición</b>	El sistema registra los datos introducidos correspondientes al cuestionario.
<b>Escenario principal</b>	
<ol style="list-style-type: none"> <li>1. El actor introduce sus datos de registro y pulsa el botón de “Continuar”.</li> <li>2. El sistema valida los datos y muestra el cuestionario de salud.</li> <li>3. El actor rellena el formulario de salud y pulsa el botón de “Aceptar”.</li> </ol>	
<b>Escenario alternativo</b>	
<ol style="list-style-type: none"> <li>3.b El usuario pulsa el botón de “Aceptar” con preguntas aún sin responder en el cuestionario.</li> <li>3.c El sistema muestra un mensaje de error por falta de información.</li> </ol>	

Cuadro 2.7: Caso de uso RF-05

<b>Caso de uso</b>	RF-05 Cerrar sesión
<b>Descripción</b>	Un usuario podrá cerrar la sesión del perfil que haya iniciado en la aplicación.
<b>Pre-condición</b>	El actor ya está registrando y validado y se encuentra en la pantalla principal de la aplicación.
<b>Post-condición</b>	La sesión del actor se ha cerrado.
<b>Escenario principal</b>	
<ol style="list-style-type: none"> <li>1. El usuario pulsa el botón de “Cerrar sesión”.</li> <li>2. El actor es redirigido a la página de inicio de sesión.</li> </ol>	

Cuadro 2.8: Caso de uso RF-06

<b>Caso de uso</b>	RF-06 Modificar perfil
<b>Descripción</b>	Un usuario podrá modificar los datos personales de su perfil, así como los datos correspondientes al cuestionario de salud.
<b>Pre-condición</b>	El actor ya está registrando y validado y se encuentra en la pantalla principal de la aplicación.
<b>Post-condición</b>	El sistema registra los datos modificados del perfil y los almacena en el sistema.
<b>Escenario principal</b>	
<ol style="list-style-type: none"> <li>1. El actor pulsa el botón de “Modificar perfil”.</li> <li>2. El sistema le muestra al usuario tanto los datos personales como los datos registrados de la encuesta de salud listos para ser modificados.</li> <li>3. El actor modifica los datos que desea.</li> <li>4. El actor pulsa el botón de “Aceptar”.</li> <li>5. El sistema lo devuelve a la pantalla inicial de la aplicación con los datos ya modificados.</li> </ol>	

*Continúa en siguiente página*



<b>Escenario alternativo</b>
<p>4.b El sistema muestra mensaje de error porque algún campo se ha dejado incompleto.</p> <p>*.b El usuario pulsa el botón de “Cancelar”.</p> <p>*.c El sistema lo devuelve a la pantalla inicial sin ninguna modificación.</p>

Cuadro 2.9: Caso de uso RF-07

<b>Caso de uso</b>	RF-07 Historial de dolores
<b>Descripción</b>	Un usuario podrá visualizar en forma de listado todos los dolores que ha añadido con anterioridad al sistema.
<b>Pre-condición</b>	El actor ya está registrando y validado y se encuentra en la pantalla principal de la aplicación.
<b>Post-condición</b>	El sistema le muestra al usuario todos los dolores que tiene registrados en la aplicación.
<b>Escenario principal</b>	
<ol style="list-style-type: none"> <li>1. El actor pulsa el botón de “Dolores” desde la pantalla principal de la aplicación.</li> <li>2. El sistema le muestra el listado de todos los dolores registrados por el usuario.</li> </ol>	
<b>Escenario alternativo</b>	
2.b El sistema muestra mensaje de error por pérdida de conexión.	

Cuadro 2.10: Caso de uso RF-08

<b>Caso de uso</b>	RF-08 Añadir un nuevo dolor
<b>Descripción</b>	Un usuario podrá añadir un nuevo registro de dolor crónico al sistema.

*Continúa en siguiente página*

<b>Pre-condición</b>	El actor ya está registrando y validado y se encuentra en la pantalla principal de la aplicación.
<b>Post-condición</b>	El sistema registra el nuevo dolor crónico añadido por el usuario.
<b>Escenario principal</b>	
<ol style="list-style-type: none"> <li>1. El actor pulsa el botón de “Nuevo Dolor” desde la pantalla del listado de dolores crónicos.</li> <li>2. El sistema le muestra la pantalla de registro de dolores crónicos.</li> <li>3. El usuario especifica la ubicación del dolor.</li> <li>4. El usuario especifica la intensidad del dolor.</li> <li>5. El usuario especifica la frecuencia del dolor.</li> <li>6. El usuario especifica el estado de actividad del dolor.</li> <li>7. El actor pulsa el botón de “Aceptar”.</li> <li>8. El sistema lo devuelve a la pantalla de listado de dolores, con el nuevo registro ya añadido.</li> </ol>	
<b>Escenario alternativo</b>	
<p>7.b El sistema muestra mensaje de error porque algún campo se ha dejado incompleto.</p> <p>7.b El sistema muestra mensaje de error por pérdida de conexión.</p> <p>*.b El usuario pulsa el botón de “Cancelar”.</p> <p>*.c El sistema lo devuelve a la pantalla de listado de dolores.</p>	

Cuadro 2.11: Caso de uso RF-09

<b>Caso de uso</b>	RF-09 Modificar registro de dolor crónico
<b>Descripción</b>	Un usuario podrá modificar cualquier registro de dolor crónico que haya añadido al sistema.

*Continúa en siguiente página*

<b>Pre-condición</b>	El actor ya está registrando y validado y se encuentra en la pantalla del listado de dolores.
<b>Post-condición</b>	El sistema registra las modificaciones realizadas sobre el dolor crónico.
<b>Escenario principal</b>	
<ol style="list-style-type: none"> <li>1. El actor pulsa el botón de “Modificar dolor” sobre el dolor registrado que desee modificar.</li> <li>2. El sistema le muestra la pantalla de modificación de dolores crónicos.</li> <li>3. El usuario modifica los valores que desea.</li> <li>4. El actor pulsa el botón de “Aceptar”.</li> <li>5. El sistema lo devuelve a la pantalla de listado de dolores, con el registro ya modificado.</li> </ol>	
<b>Escenario alternativo</b>	
<p>4.b El sistema muestra mensaje de error porque algún campo se ha dejado incompleto.</p> <p>4.b El sistema muestra mensaje de error por pérdida de conexión.</p> <p>*.b El usuario pulsa el botón de “Cancelar”.</p> <p>*.c El sistema lo devuelve a la pantalla de listado de dolores.</p>	

Cuadro 2.12: Caso de uso RF-10

<b>Caso de uso</b>	RF-10 Eliminar registro de dolor
<b>Descripción</b>	Un usuario podrá eliminar cualquier registro de dolor crónico que haya añadido al sistema.
<b>Pre-condición</b>	El actor ya está registrando y validado y se encuentra en la pantalla del listado de dolores.
<b>Post-condición</b>	El sistema eliminará el registro de dolor crónico deseado.
<b>Escenario principal</b>	

*Continúa en siguiente página*

<ol style="list-style-type: none"> <li>1. El actor pulsa el botón de “Borrar Dolor” sobre el dolor registrado que desee eliminar.</li> <li>2. El sistema lo devuelve a la pantalla de listado de dolores, sin el registro ya eliminado.</li> </ol>
--

Cuadro 2.13: Caso de uso RF-11

<b>Caso de uso</b>	RF-11 Especificar intensidad del dolor
<b>Descripción</b>	Un usuario podrá especificar la intensidad del dolor crónico padecido durante el registro del dolor o en una posible modificación del mismo.
<b>Pre-condición</b>	El actor ya está registrando y validado y se encuentra registrando o modificando un registro de dolor.
<b>Post-condición</b>	El sistema almacenará la intensidad del dolor crónico especificada.
<b>Escenario principal</b>	
<ol style="list-style-type: none"> <li>1. El actor selecciona la intensidad del dolor en la escala cromática, representando la magnitud del dolor.</li> <li>2. El usuario pulsa el botón de “Aceptar”.</li> <li>3. El sistema registra la intensidad del dolor.</li> </ol>	
<b>Escenario alternativo</b>	
<ol style="list-style-type: none"> <li>1.b El usuario pulsa el botón “Aceptar” dejando este campo sin marcar.</li> <li>1.c El sistema muestra un mensaje de error por falta de campos por rellenar.</li> <li>*.b El usuario pulsa el botón de “Cancelar”.</li> <li>*.c El sistema lo devuelve a la pantalla de listado de dolores.</li> </ol>	

Cuadro 2.14: Caso de uso RF-12

<b>Caso de uso</b>	RF-12 Especificar estado del dolor
<b>Descripción</b>	Un usuario podrá especificar el estado de actividad del dolor crónico padecido durante el registro del dolor o en una posible modificación del mismo.
<b>Pre-condición</b>	El actor ya está registrando y validado y se encuentra registrando o modificando un registro de dolor.
<b>Post-condición</b>	El sistema almacenará el estado de activación del dolor crónico especificada.
<b>Escenario principal</b>	
<ol style="list-style-type: none"> <li>1. El usuario selecciona el estado que se adecúe mejor al dolor.</li> <li>2. El usuario pulsa el botón de “Aceptar”.</li> <li>3. El sistema registra el estado del dolor.</li> </ol>	
<b>Escenario alternativo</b>	
<ol style="list-style-type: none"> <li>1.b El usuario pulsa el botón “Aceptar” dejando este campo sin marcar.</li> <li>1.c El sistema muestra un mensaje de error por falta de campos por rellenar.</li> <li>*.b El usuario pulsa el botón de “Cancelar”.</li> <li>*.c El sistema lo devuelve a la pantalla de listado de dolores.</li> </ol>	

Cuadro 2.15: Caso de uso RF-13

<b>Caso de uso</b>	RF-13 Especificar lugar del dolor
<b>Descripción</b>	Un usuario podrá especificar en la zona exacta en la que padece un dolor crónico durante el registro del dolor o en una posible modificación del mismo. Para esto el usuario deberá plasmar en un texto la descripción más detallada posible del dolor padecido.
<b>Pre-condición</b>	El actor ya está registrando y validado y se encuentra registrando o modificando un registro de dolor.
<b>Post-condición</b>	El sistema almacenará el lugar del dolor crónico especificada.
<b>Escenario principal</b>	
<ol style="list-style-type: none"> <li>1. El usuario escribe una descripción detallada del dolor.</li> <li>2. El usuario pulsa el botón de “Aceptar”.</li> <li>3. El sistema registra el lugar exacto donde se padece el dolor.</li> </ol>	
<b>Escenario alternativo</b>	
<ol style="list-style-type: none"> <li>1.b El usuario pulsa el botón “Aceptar” dejando este campo sin rellenar.</li> <li>1.c El sistema muestra un mensaje de error por falta de campos por rellenar.</li> <li>*.b El usuario pulsa el botón de “Cancelar”.</li> <li>*.c El sistema lo devuelve a la pantalla de listado de dolores.</li> </ol>	

Cuadro 2.16: Caso de uso RF-14

<b>Caso de uso</b>	RF-14 Generar informe de dolores
<b>Descripción</b>	Un usuario podrá generar de una forma automática un informe que recogerá todos los datos personales del usuario, sus respuestas en el formulario de salud y la totalidad de los dolores crónicos registrados en el sistema por el mismo (se necesita al menos un dolor crónico para poder crear el informe).
<b>Pre-condición</b>	El actor ya está registrando y validado y se encuentra en la pantalla principal de la aplicación.
<b>Post-condición</b>	El sistema generará el informe y se suministrará al usuario en forma de PDF quedando almacenado en el sistema.
<b>Escenario principal</b>	
<ol style="list-style-type: none"> <li>1. El actor pulsa el botón “Generar Nuevo Informe”.</li> <li>2. El sistema crea de forma automática e instantánea el informe.</li> <li>3. El usuario se mueve a la carpeta “Documentos” de su dispositivo y visualiza el archivo.</li> </ol>	
<b>Escenario alternativo</b>	
<ol style="list-style-type: none"> <li>1.b El sistema muestra un mensaje de error por falta de dolores crónicos almacenados.</li> </ol>	





# Capítulo 3

## Diseño

### 3.1. Arquitectura

Como se ha mencionado anteriormente en la presentación de las tecnologías a utilizar presente en el apartado 1.5, se ha utilizado SQLite como único gestor de la base de datos.

Con esto obtenemos un sistema que abandona la idea de una estructura cliente-servidor, ya que la aplicación trabajará únicamente con la base de datos local creada en el dispositivo en el que se ejecuta la aplicación, sin intervenir ningún tipo de servidor ni otro sistema gestor de base de datos.

Esta elección es debida a la simpleza que presenta la base de datos en este sistema, ya que, como veremos en los próximos apartados, solo contaremos con 3 simples entidades que no presentarán ningún tipo de reto para la ejecución de la aplicación.

Por esto podemos hablar de que en este sistema solo nos encontraremos con la capa cliente, que estará formada por los usuarios que se descarguen la aplicación en su terminal móvil Android, desde el cuál ejecutarán la aplicación e interactuarán de forma directa con la base de datos relacional creada en el dispositivo.

## 3.2. Diseño de la base de datos

En las etapas más tempranas del desarrollo del software siempre está presente el diseño de la base de datos que dará forma al sistema, ya que este será la base sobre la que gire la aplicación, siendo un espejo del funcionamiento final que tendrá el sistema.

Como es de esperar, al utilizar SQLite como SGBD, nos encontraremos ante una base de datos relacional [12].

Con esto podemos afirmar que la base de datos organizará la totalidad de datos en un conjunto de tablas, permitiendo el acceso a los datos de una forma rápida y estructurada. Además, este sistema gestor de la base de datos nos asegura la integridad de los datos y un control en el acceso a los mismos.

### 3.2.1. Diagrama Entidad-Relación

Este diagrama nos permitirá ver las entidades o tablas que estarán presentes en el sistema, así como las relaciones que existirán entre las mismas.

El Diagrama E/R del presente proyecto es el siguiente:

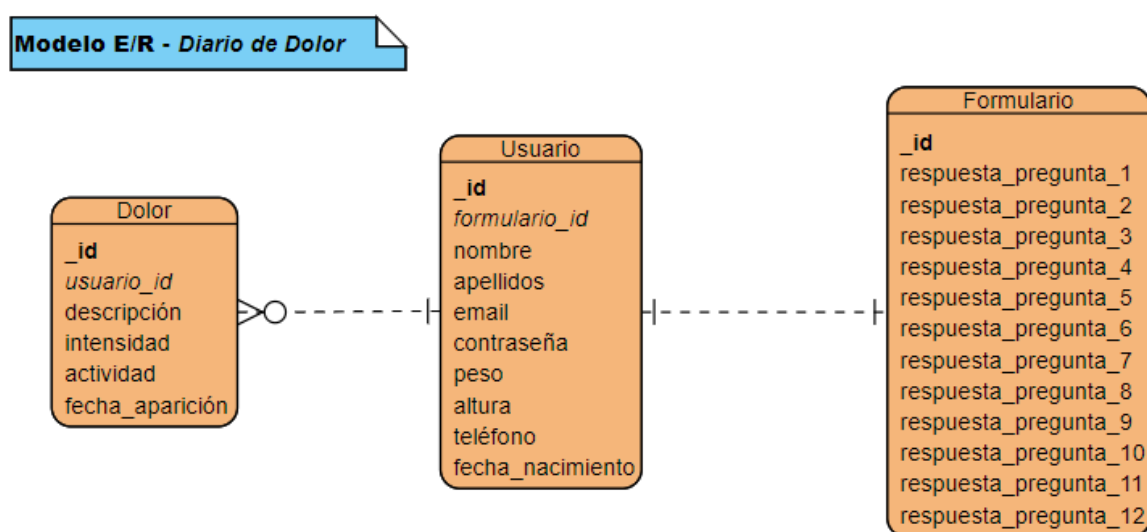


Figura 3.1: Diagrama Entidad-Relación - Diario de Dolor

### 3.2.2. Entidades y Relaciones de la base de datos

Las entidades o tablas son la parte fundamental sobre la que se basan los modelos de bases de datos Entidad/Relación, ya que son las que, al fin y al cabo, dan la estructura a la base de datos.

Por otro lado, las relaciones o uniones que conectan las tablas nos permitirán conocer con mayor medida el funcionamiento que tendrá la base de datos en la aplicación.

En este caso vemos como nos encontramos ante una estructura simple, con únicamente 3 entidades, de las que se puede destacar lo siguiente:

- **Usuario:** se podría considerar como la tabla base o principal sobre la que gira la bd. En ella se almacenan los datos personales de los usuarios que se registran en la aplicación. Además, consta con el identificador del formulario de salud rellenado por el usuario, de modo que cada usuario tendrá un único formulario ligado y que cada formulario pertenecerá a un único usuario.
- **Formulario:** entidad que recoge las respuestas rellenadas por el usuario a las preguntas correspondientes al formulario de salud. Todas estas preguntas tendrán como respuestas posibles *Si* o *No*, por lo que, al prescindir SQLite del tipo de dato *booleano*, se representará en la base de datos con un 0 para las respuestas negativas y un 1 para las respuestas afirmativas.
- **Dolor:** tabla que representará los dolores crónicos ingresados, conteniendo todos los datos correspondientes a los mismos. Cada dolor llevará consigo el identificador del usuario que lo registró, permitiendo de este modo que cada usuario pueda registrar varios dolores crónicos, pero cada uno de estos solo pertenezca a un usuario.

### 3.2.3. Diagrama de clases

Con el diagrama de clases [13], además de ver la estructura y el comportamiento que tendrá la base de datos, podremos observar más detenidamente el tipo de relación estática que tendrán las tablas y las operaciones o funciones que acompañarán a los objetos de cada clase.

El Diagrama de clases del presente proyecto es el siguiente:

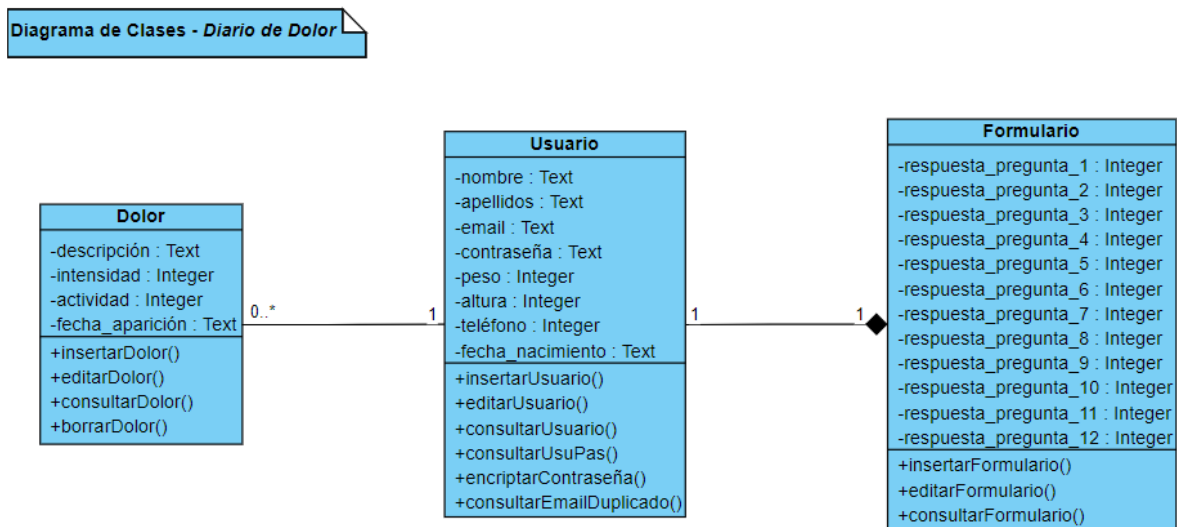


Figura 3.2: Diagrama Clases - Diario de Dolor

### 3.2.4. Operaciones y Relaciones estáticas de la base de datos

- **Operaciones:** son las funciones que caracterizan los objetos de cada clase. En cada entidad encontraremos las suyas correspondientes.

**Usuario:** como funciones básicas se podrá insertar, editar y consultar usuarios. Como se puede ver no se podrá eliminar un usuario de forma directa, pero esto será una característica innecesario para este proyecto en concreto, ya que al prescindir de la parte del servidor, únicamente encontraremos los datos del usuario en el dispositivo desde el que se ejecute la aplicación y al eliminar la aplicación se eliminará todo rastro del usuario. Además, encontramos funciones como el encriptado de la contraseña para su posterior inserción en la base de datos, la comprobación de que el correo no está siendo usado por otro usuario en el sistema y la comprobación del usuario y contraseña para la validación de acceso a la aplicación.

**Formulario:** se podrá insertar, editar y consultar un formulario. Un formulario no se podrá eliminar de forma directa ya que la entidad usuario necesita de la existencia de un formulario que complete la información correspondiente con el formulario de salud básico.

**Dolor:** se podrá insertar, editar y consultar uno o varios dolores crónicos. En este caso también se podrán eliminar los registros de dolores deseados por el usuario, ya sea porque introdujo algún dato erróneo o porque lo creó inconscientemente.

■ **Relaciones:** pasando a hablar de las relaciones estáticas del diagrama, solo nos encontramos dos relaciones:

1. **Usuario-Formulario:** en esta relación nos encontramos con que la entidad formulario de por sí misma no tendría sentido y que su contenido únicamente sirve para complementar la información ya presente en la tabla usuario. Por este motivo nos encontramos ante una **relación de composición**, que además nos afirma que el tiempo de vida de los objetos de la entidad formulario, están ligados a los de la entidad usuario.
2. **Usuario-Dolor:** relación simple de tipo **asociación** que nos muestra el enlace entre la entidad dolor y la tabla usuario, demostrando una dependencia semántica entre las mismas. También se puede ver la multiplicidad de la relación, de modo que un usuario podrá tener 0, 1 o varios dolores crónicos registrados.

### 3.3. Implementación

En el presente apartado se hará un repaso global de la implementación del proyecto, destacando los puntos y funciones más relevantes y mostrando la estructura del mismo.

Para comenzar, se explicará un poco la estructura que se ha seguido, con una división clara en dos directorios. Un primer directorio con los diseños de las pantallas (formados por archivos XML) y otro con las funcionalidades y los controladores de las mismas (con archivos Java).

El directorio *layout* con la información relativa al *frontend* de la aplicación, estará formada tanto con los diseños de las pantallas como con el diseño de la barra de herramientas. Por otro lado el directorio presente en el camino *main/java/com*, albergará todas las funcionalidades y controladores del proyecto. Esto se puede ver de una forma más clara en las siguientes imágenes.

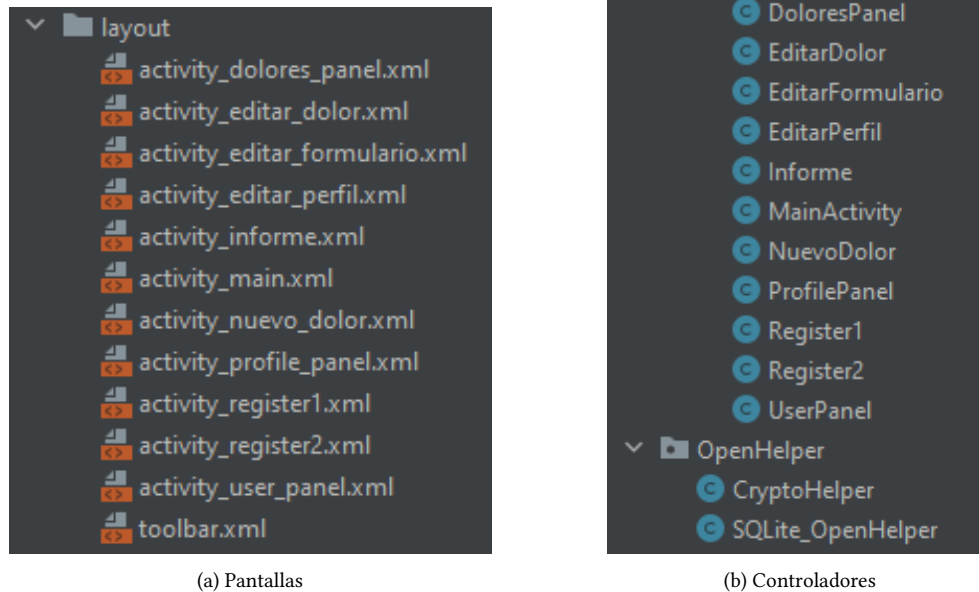


Figura 3.3: Estructura proyecto en Android Studio

Y ahora se pasará a la explicación más detallada de las implementaciones, yendo iteración por iteración mencionando lo que se ha ido añadiendo en cada una de ellas. Estas iteraciones seguirán la estructura que ya se habló en el apartado 1.6.

### 3.3.1. Implementación primera iteración

En esta primera iteración se inicia el proyecto, comenzando con la puesta a punto de la base de datos, para esto se crea un método denominado **SQLiteOpenHelper** que extiende a la clase *SQLiteOpenHelper*. En este método se implementarán todas las funciones de inserción, consultas, ediciones y eliminaciones de todas las entidades de la base de datos. Además, aquí se encontrará el método de iniciación de la base de datos, que seguirá la siguiente estructura:

```
@Override
public void onCreate(SQLiteDatabase db) {
    String query = "CREATE TABLE formulario(_ID integer primary key "+
        "autoincrement, P1 integer, P2 integer, P3 integer, P4 integer, "+
        "P5 integer, P6 integer, P7 integer, P8 integer, P9 integer, P10 "+
        "integer, P11 integer, P12 integer);";
    db.execSQL(query);
    String query2 = "CREATE TABLE usuario(_ID integer primary key "+
        "autoincrement, Email text, Password text, Nombre text, Apellidos "+
        "text, Peso integer, Altura integer, Telefono integer, "+
        "FechaNacimiento text, Formulario_ID integer REFERENCES "+
        "formulario(_ID));";
```

```

db.execSQL(query2);
String query3 = "CREATE TABLE dolor(_ID integer primary key "+
    "autoincrement, Descripcion text, FechaAparicion text, "+
    "Intensidad integer, Actividad integer, Usuario_ID integer "+
    "REFERENCES usuario(_ID));";
db.execSQL(query3);
}

```

Con la base de datos ya creada se pasó a la selección de los colores corporativos, tanto para los botones que estarán presentes en la aplicación como para la barra de herramientas que contendrá algunas pantallas. Todo esto quedará representado en la pantalla inicial del proyecto, donde ya se podrá ver la estructura y la interfaz que tendrá la aplicación.

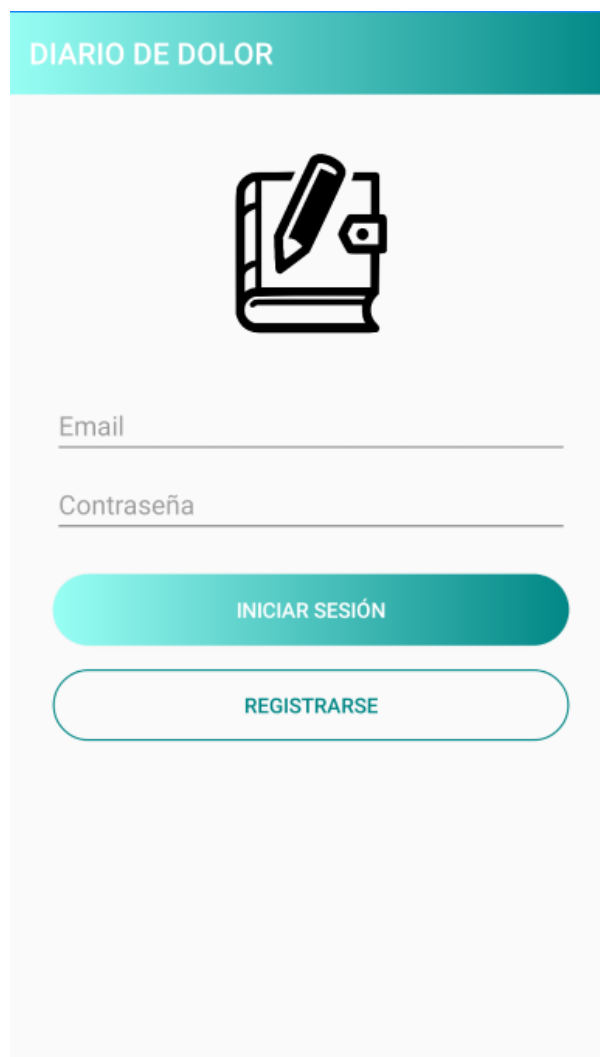


Figura 3.4: Diario de Dolor: Pantalla principal

Por otro lado, se ha implementado todo lo relacionado con el registro de nuevos usuarios, siempre con su respectivo formulario de salud. En este proceso, simplemente se recogen los datos introducidos por el usuario y se incorporan a la base de datos sin manipulación ninguna,

con la única excepción para el atributo de la contraseña, ya que no se debe introducir un dato tan privado como texto plano en la base de datos.

Este parámetro se manipulará con el algoritmo SHA-512, de modo que se agregará a la base de datos de forma ya encriptada. La elección de este algoritmo ha sido por la irreversibilidad que nos ofrece, ya que una misma cadena de caracteres cifrada con SHA-512 siempre nos dará el mismo valor generado y, a la vez, no habrá forma de a través de esta cadena cifrada, obtener el dato original introducido.

De este modo el método para comprobar las contraseñas para la validación de acceso será el siguiente:

1. Al crear el usuario se ingresa en la base de datos la contraseña ya cifrada con el algoritmo SHA-512, convirtiéndola en ilegible.
2. Para intentar acceder a la aplicación se busca si el email introducido está en la base de datos y, en caso de si estarlo, se cifra la contraseña escrita con SHA-512.
3. Se comparan las contraseñas ya cifradas, ya que si ambas proceden de la misma cadena de caracteres, el resultado obtenido tras pasarlas por el algoritmo SHA-512 será el mismo.
4. Si las claves son iguales querrá decir que se ha escrito correctamente la contraseña y se permite el acceso. En caso contrario la contraseña introducida será errónea y no se permitirá el acceso.

### **3.3.2. Implementación segunda iteración**

Pasando a la siguiente iteración, se manejará todo lo relacionado con la visualización de los datos del usuario ya ingresados durante su registro, así como la posible edición de estos mismos.

Se ha estructurado con una pantalla principal en la que se pueden visualizar los datos básicos del usuario, en la que en la parte inferior, además del botón de vuelta al menú principal, se encuentran los dos botones que llevarán, uno a la edición de los datos personales del usuario y otro a la edición de su formulario de salud.



### 3.3.3. Implementación tercera iteración

En esta iteración, ya se pasa con la implementación de la parte correspondiente a los dolores crónicos.

Con una primera pantalla principal, en la que se podrán visualizar los dolores crónicos ya ingresados por el usuario (si los hubiera), y un botón en la parte inferior de la pantalla junto al botón de vuelta al menú principal, que permite añadir nuevos dolores crónicos.

Tanto las pantallas de añadir un nuevo dolor como la de edición de los mismos seguirán un idéntico formato, solo que la correspondiente a la edición de los dolores contará con los datos del dolor ya introducidos, listos para ser editados si el usuario lo deseara, y con la adición de un nuevo botón en la parte inferior de la pantalla para eliminar de la base de datos el dolor seleccionado si así lo deseara el usuario.

DIARIO DE DOLOR

Añadir Nuevo Dolor

Descripción Lugar/Zona del Dolor

Fecha Aparición del Dolor

Intensidad del Dolor Sufrido

Actividad del Dolor

CANCELAR ACEPTAR

(a) Captura 1

DIARIO DE DOLOR

Editar Dolor

Descripción Lugar/Zona del Dolor

Fecha Aparición del Dolor

Intensidad del Dolor Sufrido

Actividad del Dolor

BORRAR DOLOR CANCELAR ACEPTAR

(b) Captura 2

Figura 3.5: Diario de Dolor: edición y creación de dolores

### 3.3.4. Implementación cuarta iteración

Para finalizar con la aplicación, primero se retocaron algunos detalles de la misma para perfeccionar su funcionamiento. Como, por ejemplo, revisar las proporciones y la ubicación de los elementos en las pantallas, o añadir funcionalidades nuevas como a la hora de añadir o editar un dolor o el propio perfil del usuario, se compruebe que todos los datos han sido rellenados, y en caso de faltar alguno que se muestre un mensaje de advertencia y no se llegue a realizar la operación de inserción o edición en la base de datos.

También se añadió la característica de que si el usuario iniciado sesión, no pulsa el botón de cerrar sesión antes de terminar la ejecución de la aplicación, a la hora de volver a abrir la aplicación ésta lo recuerde y no tenga que volver a validar sus credenciales. Esto se añadió apoyado con el paquete *SharedPreferences*, que permite guardar variables globales no volátiles.

De este modo, cuando un usuario inicia sesión se guarda una variable del tipo *SharedPreferences* recordando sus credenciales y a la hora de cerrar sesión esta variable se elimina. Por lo que al iniciarse la aplicación si esta variable tiene unas credenciales almacenadas significará que el usuario en cuestión no cerró sesión antes de cerrar la aplicación por lo que se le enviará directamente al panel de usuario. Sin embargo, si esta variable está vacía al iniciar la aplicación significará que no hay sesión que recordar y se mostrará la pantalla principal de la aplicación para el inicio de sesión.

Con esto ya se pasó a la implementación de la última funcionalidad que quedaba por cumplir de la aplicación, la creación de los informes.

Para esto lo primero que se debe implementar es la solicitud del permiso de escritura y lectura en el dispositivo, ya que el documento creado por la aplicación se debe almacenar en el mismo dispositivo donde se ejecuta. Este permiso se solicita con la siguiente línea de código:

```
ActivityCompat.requestPermissions(this,  
    new String[]{Manifest.permission.WRITE_EXTERNAL_STORAGE,  
        Manifest.permission.READ_EXTERNAL_STORAGE},  
    PackageManager.PERMISSION_GRANTED);
```

Con este permiso concedido, el usuario solo deberá pulsar el botón de **Generar nuevo informe** y la aplicación creará el documento en cuestión en la carpeta *Documentos* del dispositivo. Esto solo se llevará a cabo si el usuario ha añadido previamente algún dolor crónico al sistema, ya que de otro modo, no tendría sentido generar un informe de dolores crónicos con ningún dolor que poner en él.

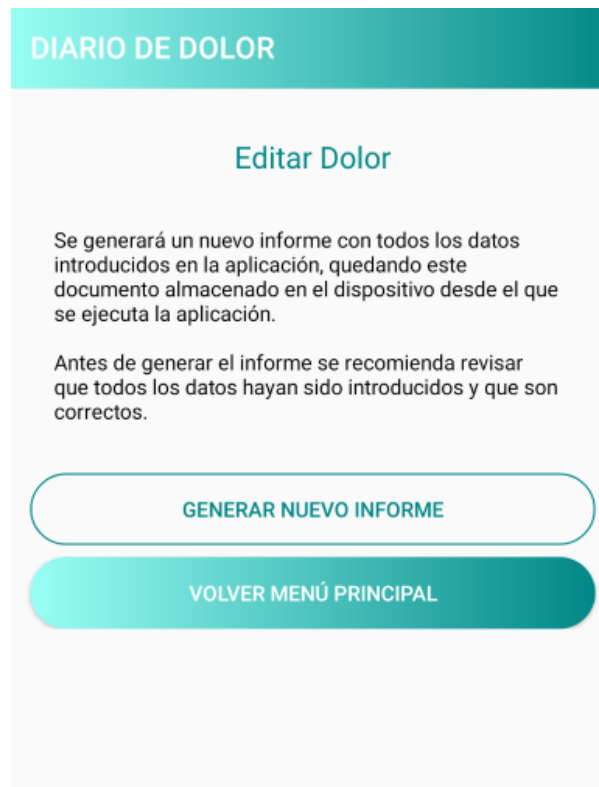


Figura 3.6: Diario de Dolor: Pantalla informes



# Capítulo 4

## Conclusión

### 4.1. Conclusiones

La realización del presente Trabajo de Fin de Grado ha acabado con un cumplimiento total de los objetivos propuestos para este proyecto, dando como resultado una aplicación totalmente funcional y operativa.

Se ha realizado la construcción de la aplicación al completo, ya que el punto de inicio del proyecto fue desde cero, por lo que se pasó por todas las etapas necesarias para la correcta ejecución del trabajo. Este proceso ha dado lugar a la aplicación **Diario de Dolor**, con la que el usuario podrá almacenar el historial de los dolores crónicos padecidos a lo largo de su vida, además de almacenar información de salud y de carácter personal que se volcarán en el informe generado para la aplicación y que podrá ser de gran utilidad para los especialistas médicos en la futura búsqueda de un posible tratamiento para los dolores.

Mencionada aplicación ha sido diseñada bajo la herramienta Android Studio, usando como lenguajes principales XML y Java. Hay que decir que las bases de ambos lenguajes de programación ya eran conocidos por los años de estudio en el grado, pero este trabajo ha permitido ampliar conocimientos y profundizar mucho más en las funcionalidades que nos ofrecen. Como ejemplo de esto se podría mencionar paquetes de Java como *SharedPreferences* o *SQLiteOpenHelper* empleados en el proyecto y que han sido totalmente imprescindibles para la correcta puesta a punto de la aplicación y que, hasta hace unos meses, eran totalmente

desconocidos para mi persona.

Además, se han empleado otras herramientas como *Visual Paradigm Online* para todo el diseño de diagramas o *Latex* para la creación de este mismo documento, que antes del inicio de este proyecto eran totalmente desconocidas para mi. En ambos casos, con el avance del proyecto, he adquirido una enorme formación en los mismos, llegando a tener una gran soltura en el manejo de los mismos.

Por todos estos motivos considero que la ejecución del presente proyecto me ha enriquecido enormemente tanto personal, por el ámbito en el que se ha movido el proyecto, como laboral, debido a los grandes conocimientos que he adquirido durante la ejecución del mismo. De este modo, en líneas generales, acabo con un muy buen sabor de boca por el trabajo realizado.

## **4.2. Líneas futuras**

El presente proyecto ha dado lugar a una aplicación móvil totalmente funcional que cumple con todos los requisitos y objetivos propuestos para este trabajo. Sin embargo, el proyecto permite algunas ampliaciones que podría añadir nuevas funcionalidades interesantes o simplemente mejorar algunos aspectos de la aplicación actual.

Como futuras líneas de trabajo se podrían destacar las siguientes:

1. Mejorar el logo de la aplicación y añadir la posibilidad de subir una imagen de perfil para los usuarios. De este modo se mejoraría la imagen de la marca y se podría añadir la foto del usuario en el informe, dándole aún más sensación de profesionalidad.
2. Implementar un modelo gráfico del cuerpo donde el usuario, al añadir un nuevo registro de dolor, pueda seleccionar el lugar donde padece el dolor. Esto se podría hacer mediante unas imágenes con la silueta del cuerpo humano, en las que el usuario pueda seleccionar el lugar exacto donde le duele. Además esta información también deberá aparecer de algún modo en los informes generados por la aplicación.
3. Implementar toda la parte del servidor, de modo que la aplicación se conecte a una base de datos remota y almacene allí toda la información. Con esto se podría añadir una

nueva tabla o entidad que almacene todos los informes generados por la aplicación. Para el envío de estos informes se recomienda el uso de algún estándar para el intercambio electrónico de información clínica como podría ser HL7.





# Bibliografía

- [1] M.M.A.A. Figueroa. Meise: Metodología de ingeniería de software. *Revista Internacional de Educación en Ingeniería ISSN*, 1116, 1940.
- [2] ALGOS Blog. Fibroline [Versión 3.0]. <https://play.google.com/store/apps/details?id=com.welltech.fibrolineseniorfree>, 2019.
- [3] Dilip Datta. *LaTeX in 24 Hours: A Practical Guide for Scientific Writing*. Obtenido en DOI: 10.1007/978-3-319-47831-9. Springer, 2017.
- [4] Wikilibros. Manual de Latex/Texto completo. [https://es.wikibooks.org/wiki/Manual\\_de\\_LaTeX/Texto\\_completo](https://es.wikibooks.org/wiki/Manual_de_LaTeX/Texto_completo), 2013, 25 dic.
- [5] Wikipedia, la Enciclopedia Libre. Lenguaje unificado de modelado. [https://es.wikipedia.org/wiki/Lenguaje\\_unificado\\_de\\_modelado](https://es.wikipedia.org/wiki/Lenguaje_unificado_de_modelado), 2021, 6 mar.
- [6] Wikipedia, la Enciclopedia Libre. Android Studio. [https://es.wikipedia.org/wiki/Android\\_Studio](https://es.wikipedia.org/wiki/Android_Studio), 2021, 9 mar.
- [7] Wikipedia, la Enciclopedia Libre. Java (lenguaje de programación). [https://es.wikipedia.org/wiki/Java\\_\(lenguaje\\_de\\_programaci%C3%B3n\)](https://es.wikipedia.org/wiki/Java_(lenguaje_de_programaci%C3%B3n)), 2021, 15 mar.
- [8] Wikipedia, la Enciclopedia Libre. Extensible markup language. [https://es.wikipedia.org/wiki/Extensible\\_Markup\\_Language](https://es.wikipedia.org/wiki/Extensible_Markup_Language), 2021, 10 sep.
- [9] Wikipedia, la Enciclopedia Libre. SQLite. <https://es.wikipedia.org/wiki/SQLite>, 2020, 5 oct.

- [10] PRESSMAN. Análisis de requisitos del software. <https://tesuva.edu.co/phocadownloadpap/Anlisis%20de%20requisitos%20del%20software.pdf>, 2002.
- [11] Marco de Desarrollo de la Junta de Andalucía (MADEJA). Guía para la redacción de casos de uso. <http://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/416#>, 2018.
- [12] Wikipedia, la Enciclopedia Libre. Base de datos relacional. [https://es.wikipedia.org/wiki/Base\\_de\\_datos\\_relacional](https://es.wikipedia.org/wiki/Base_de_datos_relacional), 2021, 12 jun.
- [13] Manuel Cillero. Diagrama de clases. <https://manuel.cillero.es/doc/metodologia/metrica-3/tecnicas/diagrama-de-clases/>, 2016.

# Apéndice

## Manual usuario

Para acceder a la aplicación no será necesaria la conexión a Internet, pero si se deberá comprobar que la aplicación está instalada en el dispositivo desde el que se intenta acceder a ella.

De este modo si la aplicación está instalada correctamente y se ejecuta, veremos la pantalla de inicio de sesión de la aplicación.

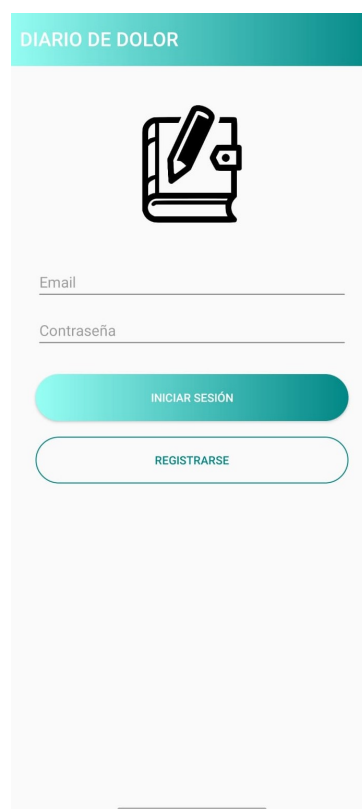


Figura 1: Diario de Dolor: Captura Pantalla Inicio de Sesión

Si es la primera vez que se ejecuta la aplicación, no existirá ningún usuario registrado, por lo que se deberá presionar el botón inferior de **Registrarse**, que nos llevará al método de registro de nuevos usuarios en la aplicación. Para esto se deberá rellenar una primera pantalla con los datos personales del usuario y al darle al botón de **Continuar** nos llevará a una nueva pantalla con el registro del formulario de salud. Todos estos datos son obligatorios e imprescindibles para la aplicación, por lo que se no se podrá crear ningún usuario hasta que se rellenen todos los datos.

### Registro de nuevo usuario

Nombre

Apellidos

Peso (kg)

Altura (cm)

Teléfono

Fecha de nacimiento

22/9/2021

Email

Contraseña

CANCELAR CONTINUAR

### Registro formulario de salud

¿Le han operado anteriormente?

¿Ha sufrido alguna reacción a la anestesia?

¿Ha tomado recientemente alguna droga: cannabis, cocaína, opiáceos u otras?

¿Tiene o ha tenido la presión arterial alta?

¿Ha padecido trombosis/flebitis/tromboflebitis?

¿Necesita más de una almohada para dormir?

¿Tiene alguna enfermedad pulmonar?

¿Tiene apnea del sueño?

¿Ha tenido convulsiones, pérdida de conocimiento, embolia o derrame cerebral?

¿Ha padecido anemia?

¿Tiene alguna objeción a la transfusión de sangre?

¿Toma aspirinas o algún otro medicamento anticoagulante?

CANCELAR REGISTRARSE

(a) Datos personales

(b) Formulario salud

Figura 2: Diario de Dolor: Capturas Registro de Usuarios

Una vez registrado, el usuario podrá iniciar sesión con los credenciales introducidos y le llevará a la pantalla principal de la aplicación. En esta pantalla podremos ver 4 botones, de modo que el situado en la parte inferior derecha será para cerrar sesión y los otros 3 botones nos llevarán a sus respectivas pantallas.

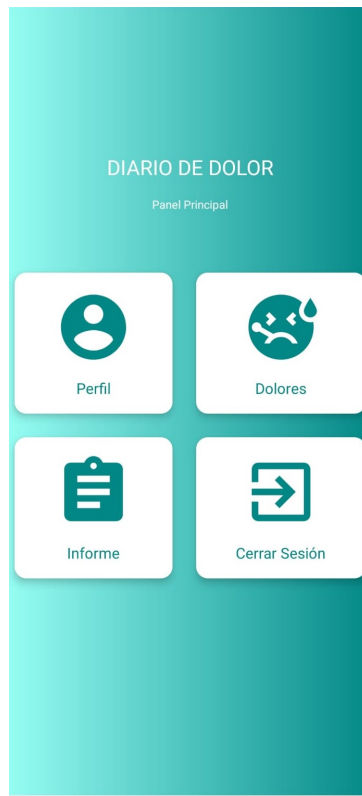


Figura 3: Diario de Dolor: Captura Panel principal

**Perfil** Este botón nos llevará a todo lo relacionado con los datos del usuario. Desde aquí se podrá tanto visualizar como editar los datos personales introducidos y el formulario de salud básico rellenido.

**Dolores** Desde esta pestaña se podrá ver el listado con todos los dolores añadidos al sistema, además de poder editar los ya introducidos o incluso añadir un nuevo dolor crónico al sistema.

**Informe** Esta pestaña nos permitirá generar el informe con todos los datos introducidos en la aplicación hasta el momento.

**Cerrar Sesión** Este botón cerrará la sesión del usuario y lo llevará a la pantalla de inicio de sesión.

En la pestaña de Perfil, nos encontraremos con los datos del usuario junto a los botones de **Editar Perfil** y **Editar Usuario** que, para cuando el usuario lo necesite, le llevará a las pantallas de edición de ambos campos.

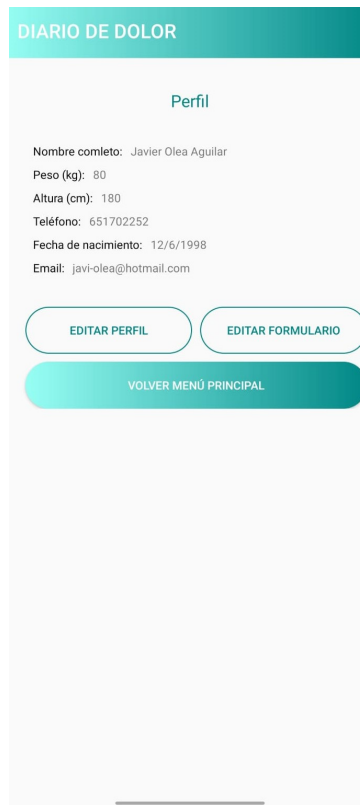


Figura 4: Diario de Dolor: Captura Perfil

Por otro lado en la pantalla de **Dolores** nos encontramos con la lista de los dolores crónicos añadidos al sistema por el usuario, y en caso de no haber añadido aún ninguno, la lista estará vacía, como es de esperar. Para añadir los dolores, se debe pulsar el botón en la parte inferior de **Nuevo Dolor**, lo que nos llevará a la pantalla de creación de los nuevos dolores.

De este modo, si ya hay dolores añadidos, al clicarlos en la lista, nos llevará a su pantalla de edición, donde se podrán modificar los datos deseados, además de poder eliminarlo del sistema si así es deseado (para esto habrá que pulsar el botón de **Borrar Dolor**).

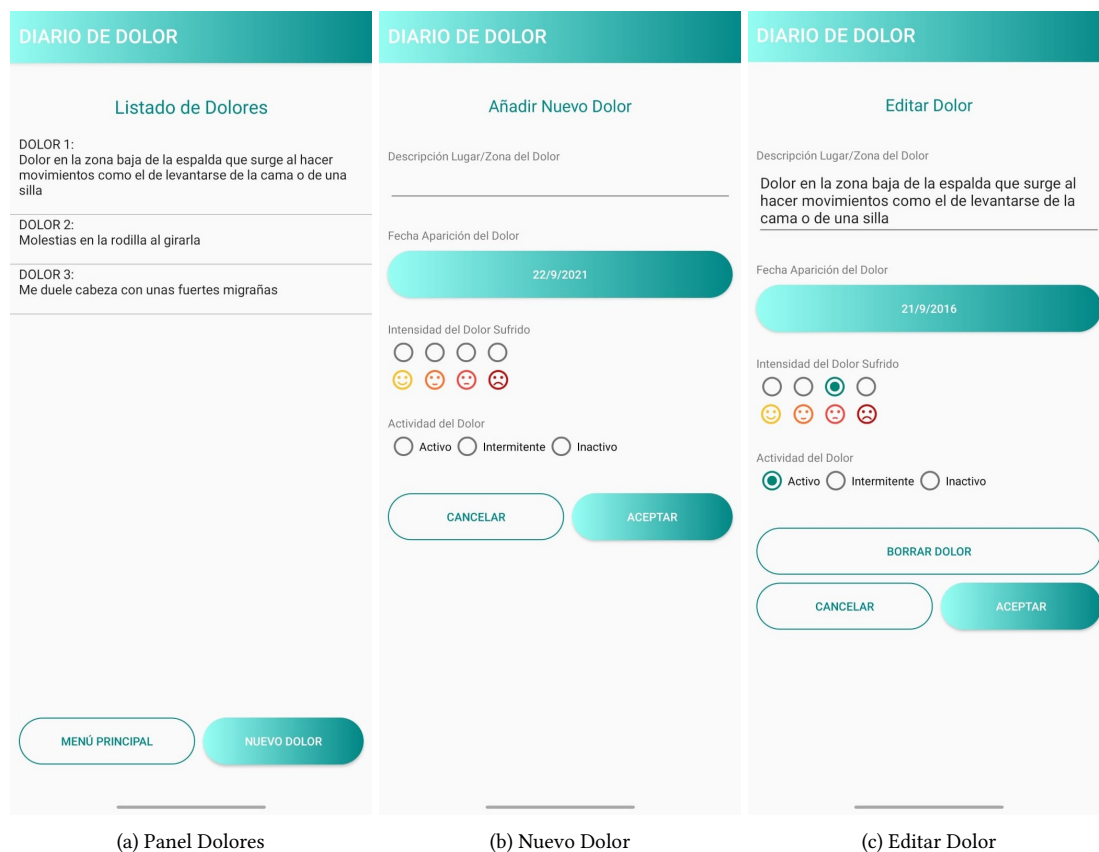
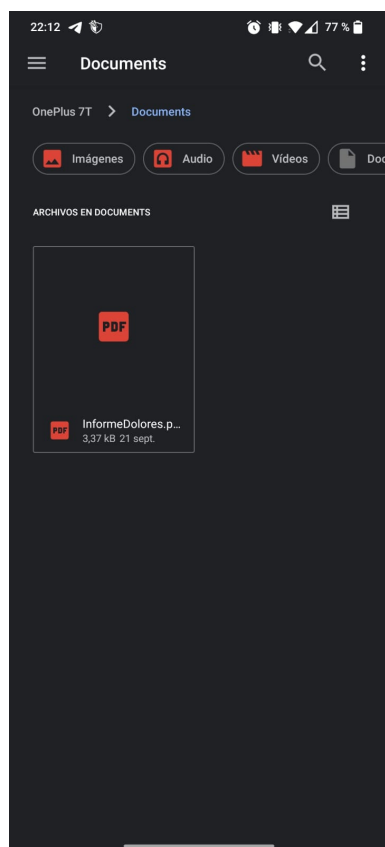


Figura 5: Diario de Dolor: Capturas Dolores

Una vez comprobado que los datos personales añadidos por el usuario son correctos y ya se han añadido la totalidad de los dolores crónicos padecidos, se puede pasar a la creación del informe de los dolores. Para ello, vamos a la pestaña de **Informes** y se pulsa el botón de generar informe.

Con esto ya se habrá creado el informe de forma automática y solo habrá que ir a la carpeta *Documentos* del sistema para abrir el informe y poder visualizarlo.



(a) Directorio creación Informe

**Informe Dolores**  
Informe generado por la aplicación 'Diario de Dolor'  
Fecha creación del informe: 21/9/2021

**DATOS DEL PACIENTE**

Nombre Completo: Javier Otero Aguilar  
Peso: 80  
Fecha de Nacimiento: 12/6/1998  
Correo Electrónico: javi-otoa@hotmail.com

Altura: 180  
Teléfono: 651702252

**FORMULARIO DE SALUD DEL PACIENTE**

Pregunta	Si	No
¿Le han operado anteriormente?	X	
¿Ha sufrido alguna reacción a la anestesia?		X
¿Ha tomado recientemente alguna droga: cannabis, cocaína, opiáceos u otros?		X
¿Tiene o ha tenido la presión arterial alta?		X
¿Ha padecido trombosis (flebitis/tromboflebitis)?	X	
¿Necesita más de una almohada para dormir?	X	
¿Tiene alguna enfermedad pulmonar?		X
¿Tiene apnea del sueño?		X
¿Ha tenido convulsiones, pérdida de conocimiento, embolia o derrame cerebral?		X
¿Ha padecido anemia?	X	
¿Tiene alguna objeción a la transfusión de sangre?		X
¿Toma aspirina o algún otro medicamento anticoagulante?	X	

**DOLORES CRÓNICOS**

Descripción	Aparición	Intensidad	Actividad
Dolor en la zona baja de la espalda que impide al hacer movimientos como el de levantarse de la cama o de una silla.	21/9/2016	3	Activo
Molestias en la rodilla al girarla.	20/7/2018	4	Intermitente
Me duele la cabeza con unas fuertes migrañas.	21/4/2018	4	Intermitente

*Nota: Los valores de la columna "Intensidad" reflejan la fuerza de los dolores padecidos, correspondiendo 1 con la intensidad más 'baja' del dolor y 4 la intensidad más 'fuerte' de referencia.*

(b) Informe

Figura 6: Diario de Dolor: Capturas Informe

Al ya contar con el informe en formato PDF en el dispositivo, éste se podrá hacer llegar al especialista médico en cuestión de la forma más cómoda para ambas partes. Este paso se podrá hacer de dos formas:

1. Enviar el archivo PDF vía correo, WhatsApp o cualquier otra red social de forma directa al especialista en cuestión.
2. Descargar el documento y llevarlo de forma física en mano al especialista.







UNIVERSIDAD  
DE MÁLAGA

| **uma.es**

E.T.S. DE INGENIERÍA INFORMÁTICA

E.T.S de Ingeniería Informática  
Bulevar Louis Pasteur, 35  
Campus de Teatinos  
29071 Málaga